

THROUGH-SILICON-VIA-AWARE PREDICTION AND PHYSICAL DESIGN FOR MULTI-GRANULARITY 3D INTEGRATED CIRCUITS

A Dissertation
Presented to
The Academic Faculty

By

Dae Hyun Kim

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
in the
School of Electrical and Computer Engineering



School of Electrical and Computer Engineering
Georgia Institute of Technology
May 2012

Copyright © 2012 by Dae Hyun Kim

THROUGH-SILICON-VIA-AWARE PREDICTION AND PHYSICAL DESIGN FOR MULTI-GRANULARITY 3D INTEGRATED CIRCUITS

Approved by:

Dr. Sung Kyu Lim, Advisor
Assoc. Professor, School of ECE
Georgia Institute of Technology

Dr. Muhannad S. Bakir
Assoc. Professor, School of ECE
Georgia Institute of Technology

Dr. Hsien-Hsin S. Lee
Assoc. Professor, School of ECE
Georgia Institute of Technology

Dr. Hyesoon Kim
Asst. Professor, College of Computing
Georgia Institute of Technology

Dr. Saibal Mukhopadhyay
Asst. Professor, School of ECE
Georgia Institute of Technology

Date Approved: March, 2012

Dedicated to my parents, Jong Wan Kim and Nam Ju Jung, for their endless love.

ACKNOWLEDGMENTS

I would like to extend special thanks to my advisor, Professor Sung Kyu Lim, for his guidance and advice during my Ph.D. study. I would also like to thank Professor Hsien-Hsin Sean Lee and Professor Saibal Mukhopadhyay for their helpful suggestions. I would like to express my thanks to Professor Muhannad Bakir and Professor Hyesoon Kim, both of whom have agreed to serve on my dissertation committee. I am also grateful to Gabriel H. Loh, Professor Linda S. Milor, and Dr. Rasit O. Topaloglu.

I would like to express my deepest thanks to all of colleagues. I especially thank Dr. Michael B. Healy, a close friend who has contributed extensive, invaluable help on my research. I also thank Mohit Pathak for numerous discussions that provided insight into my research. I would also like to thank Dr. Kiran Puttsaway and Dean L. Lewis for their practical suggestions and comments on my research. I am also grateful to Dr. Muhammad Bashir for his collaboration and friendship and to Jane Chisholm for her English teaching.

I thank GTCAD members, Dr. Faik Baskaya, Xin Zhao, Young-Joon Lee, Krit Athikulwongse, Moongon Jung, Shreepad Panth, and Taigon Song for their valuable comments and feedback. I also thank Chang Liu, Suyoun Kim, and Kaiyuan Yang.

I would like to thank MARS and STING members, Dr. Dong Hyuk Woo, Tzu-Wei Lin, Mohammad M. Hossain, and Guanhao Shen. Collaborating with them inspired me in many ways. I would also like to thank GREEN members, Jeremy Tolbert, Subho Chatterjee, Kwanyeob Chae, and Amit R. Trivedi for everything we shared in our research. I also thank Chang-Chih Chen for our collaboration.

I wish to thank my friends who supported me: Seung Yi Baek, Seung Yoon Baek, Euna Kim, Yosep Kim, Young Kwan Kim, Hye Ri Lee, Ji Hoon Lee, Sung Hee Lee, Yang Don Lee, Young Ho Lee, Ji Young Myung, Miran Myung, Eun Jeong Noh, Jung Seok Oh, Taehyun Oh, Keun Ok Park, Sang Joon Seok, Ilhyung Shin, Jeong Im Yang, Laura, Michelle, Oreta, Paran, and Seon Young. I also thank Hyun Wook Kim, Dae Hyun Park,

Kyung Hoon Lee, and Hong Il Jeon.

I would also like to extend special thanks to Julia and Se Jin Sim, who have always motivated me.

I am particularly thankful to my parents, who have always been on my side, for their love and encouragement throughout my life. I also thank my brother, Dae In Kim, and sisters, Su Mi Kim and Su Hyun Kim, for their love and support. Last but not least, I would like to thank all the professors, teachers, families, and friends who guided me to become the person that I am today.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	iv
LIST OF TABLES	x
LIST OF FIGURES	xiii
SUMMARY	xviii
CHAPTER 1 INTRODUCTION	1
1.1 Contributions	1
1.2 Structures and Benefits of 3D ICs	3
1.2.1 Structures of 3D ICs	3
1.2.2 Benefits of 3D ICs	5
1.3 Issues in 3D ICs	8
1.4 Organization	10
CHAPTER 2 THROUGH-SILICON-VIA-AWARE INTERCONNECT PREDICTION OF MULTI-GRANULARITY THREE-DIMENSIONAL INTEGRATED CIRCUITS	12
2.1 Related Work	13
2.2 Preliminaries	14
2.3 TSV-Aware 3D Wirelength Distribution Model	15
2.3.1 TSV-Aware Chip Area Model	15
2.3.2 New 3D Design Parameters	16
2.3.3 Gate-level 3D Wirelength Distribution	18
2.3.4 Block-level 3D Wirelength Distribution	21
2.4 Validation	22
2.4.1 Validation of TSV Count	22
2.4.2 3D Circuit Design Scheme	23
2.4.3 Validation of Wirelength Prediction	24
2.5 Impact Study	25
2.5.1 Impact of TSV Size and Design Parameters	26
2.5.2 Case Study	29
2.6 TSV-Aware Delay and Power Prediction Model for Buffered Interconnects in 3D ICs	31
2.6.1 TSV Resistance	32
2.6.2 TSV Capacitance	33
2.6.3 Buffer Insertion Schemes	34
2.6.4 Delay Computation	36
2.6.5 Simulation Results	38
2.7 Summary	44

CHAPTER 3	ANALYTICAL MODELING OF THROUGH-SILICON-VIA CAPACITIVE COUPLING	46
3.1	Preliminaries	47
3.1.1	TSV Formation and Die Bonding	47
3.1.2	TSV Coupling Capacitance	47
3.1.3	Basic Formulas for Capacitance Computation	49
3.2	Analytical Modeling of TSV Capacitance	52
3.2.1	TSVs with Top and Bottom Neighbors	52
3.2.2	Modeling of TSV-to-TSV Coupling Capacitance	56
3.2.3	TSVs with Top, Bottom, and Side Neighbors	57
3.2.4	Modeling of Misalignment	60
3.3	TSV Capacitance Extraction and Simulation	60
3.3.1	TSVs with Top and Bottom Neighbors	60
3.3.2	TSVs with Top, Bottom, and Side Neighbors	62
3.3.3	TSV under Misalignment	62
3.3.4	Impact of TSV Capacitance on Delay	63
3.3.5	Comparison Between TSV Coupling and MOS Capacitance	65
3.4	Analyzing More General Layouts	66
3.4.1	Capacitance Computation for Meaningful TSVs	68
3.4.2	Simulation Results	69
3.5	Summary	71
CHAPTER 4	THE DESIGN OF GATE-LEVEL 3D INTEGRATED CIRCUITS	72
4.1	Preliminaries	73
4.1.1	Maximum Allowable TSV Count	73
4.1.2	Wirelength and TSV Count Trade-Off	74
4.1.3	3D IC Design Flow	74
4.2	3D Placement Algorithm	76
4.2.1	Overview of Force-Directed Placement	76
4.2.2	Overview of a 3D Placement Algorithm	77
4.2.3	Placing Cells in 3D ICs	77
4.2.4	Placing TSVs in TSV Co-placement Scheme	79
4.2.5	Net Splitting	79
4.2.6	Pre-placing TSVs in TSV-site Scheme	80
4.3	TSV Assignment	80
4.3.1	Optimum Solution for TSV Assignment	80
4.3.2	MST-based TSV Assignment	82
4.3.3	Placement-based TSV Assignment	83
4.4	Simulation Results	84
4.4.1	Net-splitting Results	84
4.4.2	Wirelength and Runtime Comparison	85
4.4.3	Metal Layers and Silicon Area Results	87
4.4.4	On Wirelength vs # TSVs	87
4.4.5	On Wirelength and Die Area vs # Dies	88
4.4.6	TSV Co-placement vs TSV-site	88

4.5	Summary	89
CHAPTER 5 THE DESIGN OF BLOCK-LEVEL 3D INTEGRATED CIRCUITS		
	90	
5.1	3D Wirelength Metrics	91
5.1.1	3D Half-Perimeter Wirelength Based on Bounding Boxes	91
5.1.2	Subnet-based 3D Half-Perimeter Wirelength	92
5.2	Signal TSV Planning	92
5.3	Estimation of TSV Locations	93
5.3.1	Computation of a Die Span of a Steiner Point	95
5.3.2	Insertion of TSVs into and between Steiner Points	97
5.3.3	Construction of Subnets	98
5.4	TSV Assignment	99
5.4.1	Global TSV Assignment	99
5.4.2	Local TSV Assignment	100
5.5	Whitespace Manipulation	100
5.6	Simulation Results	101
5.6.1	2D Floorplanning vs 3D Floorplanning	102
5.6.2	Comparison of Signal TSV Planners	104
5.6.3	Single TSV Insertion vs. Multiple TSV Insertion	105
5.7	Summary	106
CHAPTER 6 THE IMPACT OF THROUGH-SILICON VIAS ON 3D INTEGRATED CIRCUITS		
	108	
6.1	Preliminaries	109
6.1.1	Negative Effects of TSVs	109
6.1.2	Motivation	110
6.2	Library Development Flow	111
6.2.1	Overall Development Flow	112
6.2.2	Interconnect Layers	113
6.2.3	Standard Cell Library	114
6.2.4	Comparison of 45nm, 22nm, and 16nm Libraries	114
6.3	Full-Chip 3D IC Design and Analysis Methodology	118
6.4	Simulation Results	120
6.4.1	Simulation Settings	120
6.4.2	Impact on Silicon Area	121
6.4.3	Impact on Wirelength	122
6.4.4	Impact on Performance	124
6.4.5	Impact on Power	125
6.5	Summary	126
CHAPTER 7 TOPOGRAPHY VARIATION IN 3D INTEGRATED CIRCUITS		
	127	
7.1	Motivation	128
7.1.1	Feature Density of 2D and 3D IC Layouts	128
7.1.2	The Design of 3D ICs	129

7.2	TSV Density-Driven 3D Global Placement	130
7.2.1	Force-Directed Quadratic 3D Global Placement	131
7.2.2	TSV Density-Driven 3D Global Placement	132
7.3	Simulation Results	134
7.3.1	Metal1 Density Comparison	135
7.3.2	Wirelength Comparison	137
7.3.3	Impact of Landing Pad Size	138
7.3.4	Timing and Power Comparison	138
7.3.5	Number of Fills	139
7.4	Summary	139
CHAPTER 8 CONCLUSIONS		140
REFERENCES		143
R.1	Related Publications	150

LIST OF TABLES

Table 1	Architectural performance metrics [1].	6
Table 2	Comparison between 2D and 3D designs [2].	7
Table 3	Notations	17
Table 4	Validation of our prediction on TSV count for block-level placement. . .	23
Table 5	Validation of our prediction on wirelength. $N_{DIE} = 3$, $P_{TSV,place} = 1$ and $P_{TSV,route} = 0$	23
Table 6	Impact of TSV size consideration on wirelength. $N_{DIE} : 4$, $p_{gates} : 1$, $L_{gate} : 1.37\mu m$ and $L_{TSV} : 1.37\mu m$. $B = N_{gates}/N_{DIE}$ (gate-level).	25
Table 7	Case study for early design exploration	31
Table 8	Variables affecting TSV capacitance and their effects	32
Table 9	Variation of TSV capacitance	32
Table 10	Parameters and assumptions used in this paper	33
Table 11	Comparison of the maximum delay. ‘B.I.’ means ‘ B uffer I nsertion’. BIS1 is the B uffer I nsertion S cheme 1, and BIS2 is the B uffer I nsertion S cheme 2 shown in Figure 24. ‘# B’ is the number of buffers. (Design : # gates = 40M, # dies = 4, and # signal TSVs = 8.3M).	38
Table 12	Comparison of the maximum delay and buffer counts in different circuit sizes. TSV resistance is 100Ω and TSV capacitance is 20fF. # dies = 4.	40
Table 13	Total power (cell power + interconnect power + buffer internal power). unit:W. The ratios of 3D to 2D are shown in the parentheses.	41
Table 14	Additional silicon area (in mm^2) required for buffer insertion)	41
Table 15	Variables and constants used in capacitance extraction	54
Table 16	Variable settings. C.F. means ‘capacitance function’. Series means the components are connected in series (e.g., $c_{fr,3}$ is computed by the series connection of c_{s1} and c_{s2} .)	55
Table 17	Comparison of capacitances for TSVs with wires above and below the TSVs under perfect TSV-to-TSV alignment. The computation time of our model is negligible for all the cases. W is the TSV width, S is the TSV-to-TSV spacing, H is the TSV height, and R is the runtime of Raphael in minutes.	61

Table 18	Comparison of capacitances for TSVs with wires above, below, and in the side of the TSVs under perfect TSV-to-TSV alignment. The computation time of our model is negligible for all the cases. W is the TSV width, S is the TSV-to-TSV spacing, H is the TSV height, and R is the runtime of Raphael in minutes.	61
Table 19	TSV capacitance under misalignment. M_{TSV} is the misalignment ratio. Capacitance values are reported in fF . The unit of width, spacing, and height is μm	63
Table 20	Comparison between TSV capacitance and wire capacitance. The ratio of TSV capacitance to wire capacitance is reported. Wire width is $0.2\mu m$, wire thickness is $0.36\mu m$, horizontal wire spacing is $0.2\mu m$, and vertical wire spacing is $0.3\mu m$. L is the wirelength.	64
Table 21	Delay of 3D interconnects. Schematics for this simulation are shown in Figure 39. All the delay values are scaled to the boldface case.	65
Table 22	TSV-to-TSV coupling capacitance vs. TSV MOS capacitance. These numbers do not include TSV-to-wire capacitance. w is TSV width, h is TSV height, and s is TSV-to-TSV spacing.	65
Table 23	Capacitance extraction on general layouts.	70
Table 24	Benchmark Circuits	83
Table 25	Wirelength of our 3D placement with and without net-splitting	83
Table 26	Comparison of wirelength (WL), the minimum number of metal layers (ML), runtime for placement, and total silicon area for 2D and 3D (4 dies) design for IWLS 2005 benchmarks and industrial circuits. Cell occupancy is 80%, and the number of 3D nets was set to be 3% to 5% of the number of total nets during partitioning. The numbers in parentheses are ratios to 2D.	84
Table 27	Comparison of wirelength of TSV co-placement, TSV-site placement with MST-based TSV assignment, and TSV-site placement with placement-based TSV assignment. The numbers in the parentheses are ratios to TSV co-placement.	87
Table 28	Benchmark circuits. # gates is the total number of gates in the blocks, and # nets is the total number of block-level nets.	101
Table 29	Comparison of 2D and 3D floorplanning on industrial circuits. The wirelength unit is meter. Numbers in parentheses show ratios between 3D and 2D wirelengths. The TSV diameter is $2.5\mu m$, the TSV pitch is $4.0\mu m$, and the TSV length is $20.0\mu m$	103

Table 30	Comparison of signal TSV planners. Ratios between our results and [3] (Ours/[3]) are reported.	104
Table 31	Comparison of single TSV insertion, 3D MST-based multiple TSV insertion, and 3D RST-based multiple TSV insertion.	105
Table 32	Interconnect layers of 65nm [4], 45nm [5], 32nm [6], 22nm, and 16nm process technology. The 22nm and the 16nm layers are from our prediction.	112
Table 33	Width (w) and thickness (t) of metal layers used in our 22nm and 16nm process libraries. The aspect ratio for the 22nm library is 1.8 and that for the 16nm library is 1.9.	112
Table 34	Standard cells in our 22nm and 16nm standard cell libraries.	115
Table 35	FO4 delay, standard cell heights, wire sheet resistance, and unit wire capacitance ($fF/\mu m$).	115
Table 36	Input capacitance of selected standard cells in the 45nm, the 22nm, and the 16nm libraries.	117
Table 37	Benchmark circuits.	117
Table 38	Comparison of 2D layouts.	118
Table 39	TSV-related dimensions, design rules, and TSV capacitance.	120
Table 40	Additional TSV-related statistics. “c.p.” denotes critical path.	124
Table 41	Benchmark Circuits. The number of TSVs is based on two-die implementation.	133
Table 42	Multi-pass metal1 fill insertion parameters.	134
Table 43	Comparison of minimum and maximum metal1 densities in two-die implementation with 1 \times TSV. ‘before (or after)’ denotes ‘before (or after) fill insertion’.	134
Table 44	Comparison of wirelength and metal1 densities in two-die implementation with 1 \times TSV. D denotes metal1 density of a window. (Numbers in parentheses are wirelength ratios.)	135
Table 45	Critical path delay, power, and the number of fills.	136

LIST OF FIGURES

Figure 1	Via-first TSVs and face-to-face die stacking[7].	4
Figure 2	Three types of die stacking (face-to-face, face-to-back, and back-to-back) and two types of TSVs (via-first and via-last) [8].	5
Figure 3	Lengths of longest and average interconnects vs. number of strata for $r=5$ [9].	7
Figure 4	Cost of 2D and 3D ICs. [10]	8
Figure 5	Via-last TSVs, where each TSV is surrounded by neighboring TSVs and wires. A typical size of TSV is much larger than that of global wires. . .	13
Figure 6	TSVs placed (orange square, shown in Cadence Virtuoso).	14
Figure 7	Two types of TSVs. (a) via-first (b) via-last	14
Figure 8	Three types of bonding. (a) F2F (Face-to-Face) (b) F2B (Face-to-Back) (c) B2B (Back-to-Back)	15
Figure 9	Derivation of block-level 3D wirelength distribution	21
Figure 10	3D Circuit Design Scheme	24
Figure 11	Snapshot of TSVs inserted in the topmost die of the circuit “Ind 3” (see Table 5) in Cadence SoC Encounter. There are 1186 TSVs (yellow) and 80483 standard cells (black). Die area is $592\mu m \times 592\mu m$	25
Figure 12	Snapshot of connections to TSVs in the topmost die of the circuit “Ind 3” in Cadence SoC Encounter. There are 1186 connections to TSVs among 82167 nets. Die area is $592\mu m \times 592\mu m$. The white square is a TSV cell and thick yellow line shows the connection between the TSV cell and an inverter cell.	26
Figure 13	Impact of TSV size on silicon area (A), footprint area (FP) and wirelength (WL). $r : 100$	27
Figure 14	Impact of $P_{TSV,place}$ on silicon area (A), footprint area (FP) and wirelength (WL).	27
Figure 15	Impact of $P_{TSV,route}$ on silicon area (A), footprint area (FP) and wirelength (WL).	28
Figure 16	Impact of r on silicon area (A), footprint area (FP) and wirelength (WL).	28

Figure 17	Impact of N_{DIE} on silicon area (A), footprint area (FP) and wirelength (WL).	29
Figure 18	Impact of B on wirelength (WL). $N_{\text{DIE}} : 4$	29
Figure 19	Impact of B on total wirelength of intra-block and inter-block, and TSV count. $N_{\text{DIE}} : 4$	30
Figure 20	Impact of B on silicon area (A) and footprint area (FP). $N_{\text{DIE}} : 4$	30
Figure 21	Via-first TSV and its capacitive components in face-to-back die bonding .	34
Figure 22	Buffer insertion in 2D and 3D ICs	34
Figure 23	Distance-capacitance plot in a 3D wire	35
Figure 24	Two buffer insertion schemes used in this paper and their distance-capacitance plot	36
Figure 25	Delay distribution of 3D ICs (40M gates). (a) 3D with BIS1, w/o TSV RC, (b) 3D with BIS1, with TSV RC, and (c) 3D with BIS2, with TSV RC. TSV capacitance is $5fF$	38
Figure 26	Comparison of delay in various cases. Wirelength in the x -axis does not include TSV height, and it is assumed that TSVs are evenly distributed along a 3D net. The driver size is $20\times$, and buffer insertion was not used.	39
Figure 27	TSV RC vs Delay for each buffer size. (a) $1\times$ buffer, (b) $5\times$ buffer, (c) $20\times$ buffer	39
Figure 28	Circuit size vs maximum delay. # dies = 4.	40
Figure 29	Three types of die bonding (face-to-face, face-to-back, and back-to-back) and two types of TSVs (via-first and via-last).	47
Figure 30	Left: Capacitive coupling in via-first TSV technology. Right: Capacitive coupling in via-last TSV technology.	48
Figure 31	Left: TSV RC model. Right: Simplified TSV RC model.	48
Figure 32	Capacitance of multiple wires on ground plane.	49
Figure 33	Various fringe capacitances.	50
Figure 34	Fringe capacitances when surrounding wires exist.	50
Figure 35	Multiple dielectric materials in a parallel plate capacitor.	51
Figure 36	Capacitance between two surfaces.	52
Figure 37	Capacitive components of TSVs with top and bottom neighboring wires.	53

Figure 38	Capacitive components of TSVs with top, bottom, and side neighboring wires.	58
Figure 39	Schematics for the delay simulation in Table 21. C_{TSV} is the total capacitance (the sum of TSV-to-wire coupling capacitances and TSV-to-TSV coupling capacitances) of a TSV.	64
Figure 40	An example layout of a 3D IC designed by 3D IC design methodology presented in [11]. Via-first TSVs are used and two dies are stacked with face-to-back bonding. Bright rectangles are TSV landing pads (TSVs exist inside landing pads), and dark rectangles are standard cells.	66
Figure 41	Zoom-in shot of Figure 40. Bright big rectangles are TSV landing pads (TSVs exist inside landing pads), and thin vertical lines above TSVs are metal wires.	66
Figure 42	A general layout where TSVs are placed irregularly. The capacitance of T_P is computed.	67
Figure 43	Capacitance computation for a pair of TSVs. If there exists an x - or y -overlap, C_{parallel} , $C_{\text{sw,top}}$, and $C_{\text{top,top}}$ are applied as shown in (a). If there is no overlap, C_{corner} and $C_{\text{sw,top}}$ are applied as shown in (b) and (c). . . .	68
Figure 44	Two general (= non-regular) example layouts. The total number of TSVs is eight. The electric potential of one of them (= red square) is set to V_{DD} , while that of all others are set to 0.	69
Figure 45	Via-first and via-last TSVs	73
Figure 46	Two 3D IC design flows developed in this project. (a) TSV co-placement, (b) TSV-site	74
Figure 47	Splitting 3D net into subnets (side view)	79
Figure 48	Cost computation for each combination of TSVs in three dies (side view). (a) wirelength = $2L$ for (T_1, T_6) , (b) wirelength = L for (T_3, T_6)	80
Figure 49	MST-based TSV assignment (side view)	82
Figure 50	TSV assignment based on 3D Placement (top view)	82
Figure 51	Cadence SoC Encounter snapshot of the bottommost die of Ind2 designed by TSV co-placement and TSV-site methods. Routing for 3D nets are shown in blue.	84
Figure 52	Wirelength distribution of (a) des_perf, where the die width is $572\mu\text{m}$ in 2D design and $311\mu\text{m}$ in 3D design (4 dies), (b) b19, where the die width is $762\mu\text{m}$ in 2D design and $411\mu\text{m}$ in 3D design (4 dies).	85

Figure 53	Wirelength vs # TSVs of (a) des_perf, and (b) b19 for 2D and 3D (4 dies) designs	85
Figure 54	Wirelength vs # dies of des_perf in 3D design	86
Figure 55	Die area and # TSVs of des_perf in 3D design	86
Figure 56	Wirelength metrics for a 3D net. (a) HPWL based on 2D bounding boxes. (b) HPWL based on subnet construction. d is the vertical length of a TSV.	91
Figure 57	The proposed signal TSV planning flow.	93
Figure 58	A 3D IC with via-first TSV and face-to-back die stacking.	94
Figure 59	Construction of a 3D RST. (a) Points to be connected. (b) Fixed points projected onto a 2D xy plane. (c) A 2D RSMT. (d) A 3D RST constructed from (c).	96
Figure 60	Die span diagrams. Solid dots are <i>top</i> variables and empty dots are <i>bot</i> variables. Red spans show $tTop$ and $tBot$ when the die span of s is computed. (a) $tTop (=2) < tBot (=3)$. (b) $tTop (=2) = tBot (=2)$. (c) $tTop (=2) > tBot (=1)$	98
Figure 61	Global assignment of TSVs to whitespace blocks. T_i is the i -th TSV and W_j is the j -th whitespace block. f/c in each edge denotes that f is the maximum flow capacity, and c is the cost. $C.T_{ij}$ is the wirelength when TSV T_i is assigned to whitespace block W_j . $C.W_i$ is the maximum number of available TSV slots in whitespace block W_i	99
Figure 62	Full die (top-die) and zoom-in shot of four-die block-level 3D floorplanning (Cadence Virtuoso)	102
Figure 63	Development flow of our 22nm and 16nm process and standard cell libraries.	111
Figure 64	The smallest (1×) two-input NAND gates of the 45nm [12], and our 22nm and 16nm libraries (drawn to scale).	113
Figure 65	Delay of a minimum-size inverter driving an N× inverter ($N = 1, 2, 4, 8, 16$), where both inverters are in the same process. RC parasitics are included.	116
Figure 66	Size comparison of the 4 TSVs used in our study: (a) $5\mu m$ and $0.5\mu m$ width used for 45nm technology, (b) $1\mu m$ and $0.1\mu m$ width used for 22nm technology.	121
Figure 67	Zoom-in GDSII layouts of the six types of designs studied in this paper. Each TSV is surrounded by its keep-out-zone.	122

Figure 68	Comparison of the optimized 2D designs and two-die 3D designs (BM1) in 45nm, 22nm, and 16nm technology. The x-axis shows the technology combination (the first row shows TSV diameter in μm).	123
Figure 69	Comparison of the optimized 2D designs and two-die 3D designs (BM2) in 45nm, 22nm, and 16nm technology. The x-axis shows the technology combination (the first row shows TSV diameter in μm).	123
Figure 70	3D IC designed in two dies (left) and three dies (right) using via-first TSVs and face-to-back die bonding.	128
Figure 71	Before and after filler insertion. Yellow squares denotes TSVs, pink are fillers, and light blue are M1 wires.	129
Figure 72	Variations of the maximum density and the density range in metal1 layer when only one window contains landing pads ($4.14\mu m \times 4.14\mu m$). ‘before’ (or ‘after’) denotes ‘before’ (or ‘after’) fill insertion, D_{max} (or D_{min}) denotes the maximum (or minimum) window density, and D_{TSV} denotes the density of the window containing landing pads.	130
Figure 73	Screen shots of die0 of circuit C2. Dark rectangles are standard cells, and light squares are metal1 landing pads.	131
Figure 74	ΔD of die0 of WL-driven placement (left), TSV-site placement (middle), and TSV density-driven placement (right)	133
Figure 75	Maximum density gradient of die0 of WL-driven placement (left), TSV-site placement (middle), and TSV density-driven placement (right)	135

SUMMARY

The main objective of this research is to predict the wirelength, area, delay, and power of multi-granularity three-dimensional integrated circuits (3D ICs), to develop physical design methodologies and algorithms for the design of multi-granularity 3D ICs, and to investigate the impact of through-silicon vias (TSVs) on the quality of 3D ICs. This dissertation supports these objectives by addressing six research topics. The first pertains to analytical models that predict the interconnects of multi-granularity 3D ICs, and the second focuses on the development of analytical models of the capacitive coupling of TSVs. The third and the fourth topics present design methodologies and algorithms for the design of gate- and block-level 3D ICs, and the fifth topic pertains to the impact of TSVs on the quality of 3D ICs. The final topic addresses topography variation in 3D ICs.

The first section of this dissertation presents TSV-aware interconnect prediction models for multi-granularity 3D ICs. As previous interconnect prediction models for 3D ICs did not take TSV area into account, they were not capable of predicting many important characteristics of 3D ICs related to TSVs. This section will present several previous interconnect prediction models that have been improved so that the area occupied by TSVs is taken into account. The new models show numerous important predictions such as the existence of the number of TSVs minimizing wirelength.

The second section presents fast estimation of capacitive coupling of TSVs and wires. Since TSV-to-TSV and TSV-to-wire coupling capacitance is dependent on their relative locations, fast estimation of the coupling capacitance of a TSV is essential for the timing optimization of 3D ICs. Simulation results show that the analytical models presented in this section are sufficiently accurate for use at various design steps that require the computation of TSV capacitance.

The third and fourth sections present design methodologies and algorithms for gate- and block-level 3D ICs. One of the biggest differences in the design of 2D and 3D ICs is that

the latter requires TSV insertion. Since no widely-accepted design methodology designates when, where, and how TSVs are inserted, this work develops and presents several design methodologies for gate- and block-level 3D ICs and physical design algorithms supporting them. Simulation results based on GDSII-level layouts validate the design methodologies and present evidence of their effectiveness.

The fifth section explores the impact of TSVs on the quality of 3D ICs. As TSVs become smaller, devices are shrinking, too. Since the relative size of TSVs and devices is more critical to the quality of 3D ICs than the absolute size of TSVs and devices, TSVs and devices should be taken into account in the study of the impact of TSVs on the quality of 3D ICs. In this section, current and future TSVs and devices are combined to produce 3D IC layouts and the impact of TSVs on the quality of 3D ICs is investigated.

The final section investigates topography variation in 3D ICs. Since landing pads fabricated in the bottommost metal layer are attached to TSVs, they are larger than TSVs, so they could result in serious topography variation. Therefore, topography variation, especially in the bottommost metal layer, is investigated and two layout optimization techniques are applied to a global placement algorithm that minimizes the topography variation of the bottommost metal layer of 3D ICs.

CHAPTER 1

INTRODUCTION

The ever increasing demand for high performance integrated circuits (ICs) has led academia and industry to develop a variety of new technologies for faster devices, more dense integration of transistors, and faster signal transmission. Among the various promising technologies, such as graphene, on-chip optical interconnects, and extreme ultraviolet lithography, the three-dimensional (3D) IC is expected to provide extremely high chip-to-chip bandwidth and achieve higher performance than traditional two-dimensional (2D) ICs. The 3D IC is also expected to be a near-future technology because fabrication of 3D ICs requires just a few additional manufacturing processes in addition to the current silicon process. Therefore, considerable interest has arisen in developing manufacturing technologies for 3D ICs such as through-silicon via (TSV) fabrication and die-to-die bonding, 3D IC design methodologies and algorithms such as 3D placement and 3D routing, and 3D IC analysis methodologies and algorithms such as 3D timing analysis and die-to-die coupling analysis.

This dissertation focuses on interconnect prediction models that predict the quality of 3D ICs, analytical models for TSV coupling capacitance that enable fast estimation of TSV capacitance, development of design methodologies and algorithms for multi-granularity 3D ICs, investigation of the impact of TSVs on the quality of 3D ICs, and topography variation of 3D ICs. This chapter begins by listing the contributions of this research, discusses the structures, potential benefits, and issues in 3D ICs, and presents the organization of this dissertation.

1.1 Contributions

The contributions of this research are as follows.

- TSV-aware interconnect prediction models for gate- and block-level 3D ICs are developed. Previous works on interconnect prediction models of 3D ICs ignore the

TSV size, so their models do not accurately model the interconnects of 3D ICs. In addition, their models fail to identify important relationships among the number of TSVs, chip area, wirelength, and so on. Taking TSV area and TSV capacitance into the prediction models presented in this dissertation not only enables more accurate prediction of the wirelength, area, delay, and power of 3D ICs but also provides new prediction results that other interconnect prediction models ignoring TSV area and TSV capacitance cannot provide.

- Although TSV coupling capacitance can be computed by existing capacitance extraction tools, such tools require extensive computation time, so they cannot be used in computer-aided design (CAD) for design optimization, which requires extremely fast estimation of TSV coupling capacitance. Therefore, analytical models that estimate the coupling capacitance of via-first and via-last TSVs are developed and validated in this dissertation. Computation techniques for estimating the coupling capacitance of non-uniformly-placed TSVs are also developed. These analytical models and computation techniques enable fast estimation of TSV coupling capacitance, which can be used in various design steps such as the floorplanning, placement, and routing of 3D ICs.
- The design of 3D ICs requires TSV insertion and routing. However, because previous work does not physically insert TSVs into layouts, the layouts are not realistic. In addition, their simulation results such as area and wirelength are too optimistic because the negative effects of TSVs are ignored. Therefore, two design methodologies that take TSV insertion and routing into account, a 3D placement algorithm, and two TSV assignment algorithms are developed in this dissertation for the design of gate-level 3D ICs. These design methodologies and algorithms physically insert TSVs into layouts while minimizing wirelength. Various studies on the impact of TSV size and capacitance on the area, wirelength, delay, and power of 3D ICs are

also presented.

- Design methodologies and algorithms that enable the design of block-level 3D ICs are developed. A new, accurate wirelength metric is also presented for multiple TSV insertion, and a 3D rectilinear Steiner tree construction algorithm is developed to minimize wirelength. Block-level 3D ICs designed by these methodologies and algorithms have much shorter wirelength than those designed by other design methodologies and algorithms presented in the literature.
- Both TSVs and devices are becoming scaled down, so future 3D ICs will very likely be built with advanced process technologies and smaller TSVs. However, the impact of current and future TSVs on current and future 3D ICs has not been thoroughly studied. Therefore, a study pertaining to the impact of TSVs and device technologies on the quality of current and future 3D ICs is conducted and presented in this dissertation. Very detailed analyses based on GDSII-level 3D IC layouts generated by commercial software with add-on tools developed for 3D ICs show convincing results on the area, wirelength, delay, and power of current and future 3D ICs.
- Large TSV landing pads used in 3D IC layouts could result in serious density mismatch, which is a source of topography variation. However, no studies have examined how large TSV landing pads will affect topography variation in 3D ICs. Therefore, in this dissertation, topography variation in 3D ICs is investigated, and a 3D global placement algorithm that minimizes topography variation is developed.

1.2 Structures and Benefits of 3D ICs

1.2.1 Structures of 3D ICs

3D ICs are built by stacking multiple dies, illustrated in Figure 1. In the figure, each die is fabricated separately, and the two dies are aligned and bonded. When each die is fabricated, both TSVs and devices are fabricated. TSVs, which are made of conductors such as copper

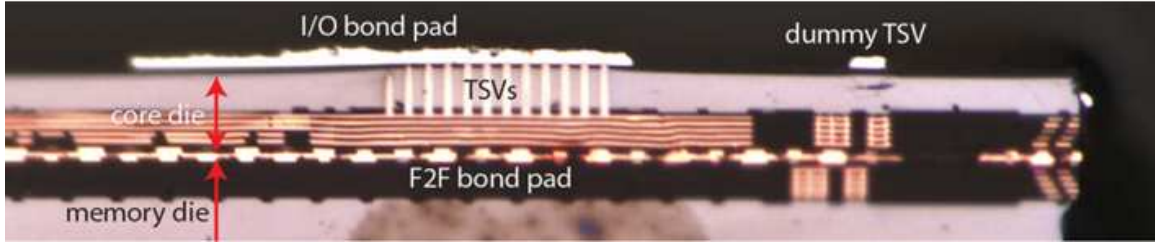


Figure 1. Via-first TSVs and face-to-face die stacking[7].

and aluminum, are used to electrically connect devices in different dies. TSVs come in two different types:¹:

- **Via-first TSVs:** Via-first TSVs are fabricated before front-end-of-line (FEOL). Since interconnect layers are deposited after via-first TSVs and devices are fabricated, via-first TSVs are connected to the bottommost metal layer. Metal pieces attached to TSVs are called TSV landing pads, which are also fabricated on the back side of the silicon substrate. These back-side TSV landing pads of a die are connected to the back-side TSV landing pads, or the topmost metal layer of the other die.
- **Via-last TSVs:** Via-last TSVs are fabricated after back-end-of-line (BEOL). Since via-last TSVs are fabricated through whole layers (both the silicon and interconnect layers), TSV landing pads are fabricated at both ends of via-last TSVs.

Figure 2 illustrates via-first and via-last TSVs. TSV size is constrained by the aspect ratio (TSV width:TSV height). As a result, via-first TSVs are usually smaller than via-last TSVs, so the former occupy a smaller area than the latter. In addition, via-last TSVs occupy interconnect layers, so they are expected to cause higher routing congestion than via-first TSVs.

After the fabrication of each die, multiple dies are stacked and bonded sequentially and if necessary, they are thinned. Stacking is categorized into three types:

- **Face-to-face (F2F) stacking:** The front sides of two dies are bonded, illustrated in Figure 2(a). In this case, TSVs are not used to connect the two dies. Instead, the

¹Via-middle TSVs are not discussed in this dissertation.

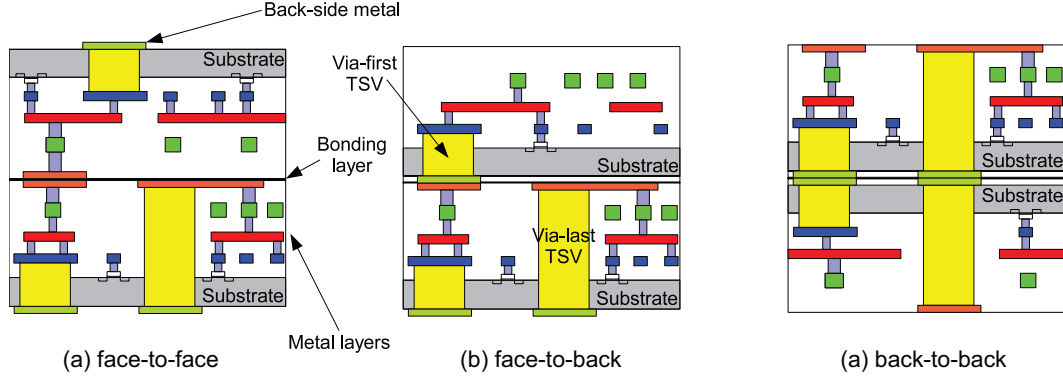


Figure 2. Three types of die stacking (face-to-face, face-to-back, and back-to-back) and two types of TSVs (via-first and via-last) [8].

topmost metal layer is used as bondpoints.² In this case, die-to-die communication is realized through local wires and vias.

- Face-to-back (F2B) stacking: The front side of a die and the back side of the other die are bonded, illustrated in Figure 2(b). In this case, a signal goes through only one TSV when it is transmitted from one die to another die.
- Back-to-back (B2B) stacking: The back sides of two dies are bonded, illustrated in Figure 2(c). In this case, a signal goes through two TSVs.

1.2.2 Benefits of 3D ICs

3D ICs are expected to provide numerous benefits over 2D ICs as follows:

- Extremely high bandwidth: The bandwidth between separate chips is limited by the number of I/O pins that each chip can have. However, dies vertically stacked in a 3D IC can use TSVs as their communication channels, so the number of connections between dies in 3D ICs is not limited [1, 13]. Table 1 illustrates extremely high bandwidth obtained in a 3D processor. For comparison, the maximum bandwidth of single data rate synchronous dynamic random-access memory (SDR SDRAM) that operates at 277 MHz and transfers 8 bytes at a time is about 2.2 GB/s. However,

²Numerous bonding technologies have been developed. For example, several technologies use micro-bumps between the topmost metal layers. This work, however, examines only direct metal-to-metal bonding.

Table 1. Architectural performance metrics [1].

Benchmark	Memory Bandwidth (GB/s)	IPC per core	BIPS
string_search	8.9	0.65	11.52
matrix_multiply	13.8	0.32	5.67
median	63.8	1.62	28.72
aes_encrypt	49.5	0.97	17.20
motion estimation	24.1	1.20	21.27
histogram	30.3	0.90	15.96
edge detection	15.6	0.95	16.84
k-means	40.6	0.94	16.66

the maximum bandwidth of the memory structure, which operates at 277 MHz and transfers 4 bytes at a time, used in [1], is about 71 GB/s, and the memory bandwidth simulated for each benchmark ranges from 8.9 GB/s to 63.8 GB/s as shown in Table 1.

- **Performance improvement:** If a design is implemented in 3D, the footprint area of the chip becomes smaller. This smaller footprint area reduces the average gate-to-gate or block-to-block wirelength [9, 14]. Therefore, 3D ICs are expected to achieve better performance than 2D ICs. For example, [9] shows predicted wirelength reduction in 3D ICs (Figure 3). As the figure illustrates, as more dies are stacked, the average wirelength and the corner-to-corner wirelength decrease and the longest wirelength also decreases. Table 2 shows performance improvement obtained by 3D ICs. In the table, the total wirelength decreases from 19.107 m to 8.238 m (56.9% improvement) and the maximum speed increases from 63.7 MHz to 79.4 MHz (24.6% improvement).
- **Low power:** Shorter wirelength results in lower dynamic power consumption [15, 16]. In addition, shorter wirelength also leads to lower gate switching power because smaller gates can be used to drive smaller wire capacitance. Table 2, from [2], shows the power reduction obtained by 3D ICs. Moreover, if multiple chips are integrated in a 3D IC, the I/O power used for inter-chip signal transmission can be significantly

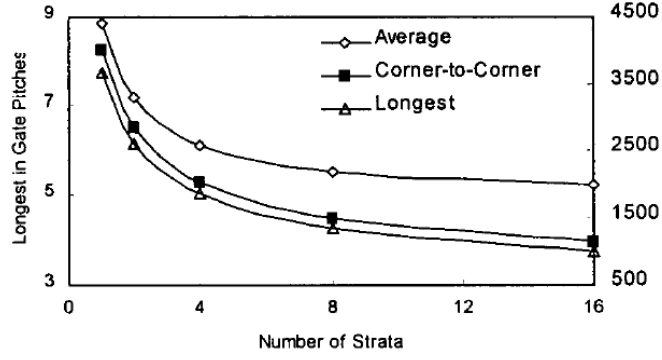


Figure 3. Lengths of longest and average interconnects vs. number of strata for $r=5$ [9].

Table 2. Comparison between 2D and 3D designs [2].

Metric	2D	3D	%
Total Area (mm^2)	31.36	23.40	25.3
Core Area (mm^2)	29.16	20.16	30.9
Mean Net Length (μm)	836.0	392.9	53.0
Total Wire Length (m)	19.107	8.238	56.9
Max Speed (MHz)	63.7	79.4	24.6
Critical Path (ns)	15.7	12.6	19.7
Logic Power 63.7 MHz (mW)	340.0	324.9	4.4
Logic Power 79.4 MHz (mW)	-	409.2	-
FFT Logic Energy (μJ)	3.552	3.366	5.2

reduced [17].

- Smaller form factor: The form factor is a very important constraint for chips that require as small area as possible, such as bio chips. If blocks and gates are placed in multiple dies and the dies are stacked, the footprint area significantly decreases [10]. Ideally (assuming that the TSV size is zero), the footprint area of a 3D IC with N dies is $1/N$ of that of its 2D counterpart.
- Cost: 3D ICs are expected to have lower cost for large designs [10]. Figure 4 shows the cost of 3D ICs [10]. In the figure, the cost of 3D ICs for large designs (more than 100 M gates) is lower than that of 2D ICs. The reason that 2D ICs are more costly than 3D ICs for large designs is due to the exponential relationship between the die size and the yield. Since the area of each die of a 3D IC is much smaller than that of

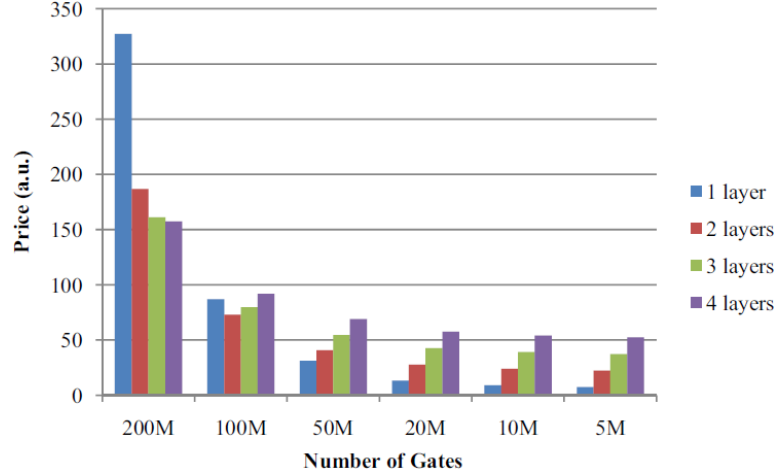


Figure 4. Cost of 2D and 3D ICs. [10]

its 2D counterpart, 2D ICs could cost much more than 3D ICs for large designs.

- **Heterogeneous integration:** Integrating different process technologies in a single chip is very difficult. Therefore, fabricating each die using its own process technology and integrating the dies in a 3D IC provide the integration of various circuit components in a single chip. For example, LSI, MEMS, and optoelectronics devices are integrated and connected by TSVs in a single chip [18],

1.3 Issues in 3D ICs

Although 3D ICs provide many benefits, many issues pertaining to them should be analyzed, studied, and resolved. First, TSVs should be placed and routed. Since TSVs occupy silicon area, ignoring TSV size and locations in the design of 3D ICs leads to the generation of unrealistic layouts and an underestimation of chip area, wirelength, and routing congestion. However, as of early 2012, no standardized 3D IC design methodology was yet available, calling for development of new physical design methodologies and algorithms handling TSVs. Second, physical phenomena related to TSVs should also be included in the analysis of 3D ICs. For example, TSV capacitance is an important source of delay degradation in 3D ICs, TSV-to-TSV coupling capacitance is a non-negligible factor in the

signal integrity analysis, and stress caused by TSVs impacts carrier mobility in 3D ICs. Although several existing analysis tools for 2D ICs can handle 3D IC structures, analysis tools that can natively analyze 3D ICs are still needed. Other issues such as die alignment, die-to-die bonding, yield, packaging, thin die handling, and so on must also be addressed.

This dissertation focuses on issues pertaining to the prediction of the quality of 3D ICs, the capacitance of TSV-to-TSV coupling, design methodologies for multi-granularity 3D ICs, algorithms enabling design methodologies, the impact of TSVs on the quality of 3D ICs, and topography variation in 3D ICs.

- Interconnect prediction models: Analytical models predicting the interconnects of 3D ICs provide a fast estimation of area, wirelength distribution, average wirelength, delay, and power consumption of 3D ICs. These models enable the exploration of very large 3D IC design space.
- TSV coupling capacitance: TSVs have non-negligible capacitance, so fast estimation of TSV capacitance must be incorporated into 3D IC design algorithms such as 3D placement and 3D routing. Analytical models computing the TSV coupling capacitance enable 3D IC design tools to use accurate TSV capacitance during the design and optimization of 3D ICs.
- Design methodologies for multi-granularity 3D ICs: Design methodologies and physical design automation algorithms for 3D ICs are essential to the development of 3D ICs. Although several existing design methodologies and algorithms for 2D ICs can be applied to the design of 3D ICs, the development of 3D ICs necessitates TSV-aware 3D IC design methodologies and algorithms.
- The impact of TSVs on the quality of 3D ICs: The various TSV sizes and capacitances affect the quality of 3D ICs in different ways. Device technologies also have a strong impact on the quality of 3D ICs. Therefore, an investigation of the impact of

both TSV technologies and device technologies on 3D ICs can provide a quantitative and qualitative assessment of current and future 3D ICs.

- Topography variation in 3D ICs: Topography variation has been one of the most critical issues for the design of 2D ICs for manufacturability. Since 3D ICs use large TSV landing pads, the impact of TSV landing pads on topography variation in 3D ICs must be investigated.

1.4 Organization

This dissertation is organized as follows:

- In Chapter 2, an existing non-TSV-aware wirelength prediction model is improved to account for the effects of TSVs on 3D ICs. This TSV-aware wirelength prediction model is also extended and combined with buffer insertion schemes for the prediction of the delay and power of 3D ICs. Validation of the models is provided and various prediction results are presented.
- In Chapter 3, analytical models that estimate the coupling capacitance of TSVs are developed. The first analytical model estimates the capacitance of a TSV in a uniformly-placed TSV array. The second analytical model estimates the capacitance of a TSV in a non-uniformly-placed TSV array, a more general structure than the uniformly-placed TSV array. Capacitance values, a breakdown of the values, and runtime of the analytical models are compared with those of the simulation using a commercial tool.
- In Chapter 4, two design methodologies for gate-level 3D IC design, TSV co-placement and TSV site methodologies, are presented. The TSV co-placement design methodology places TSVs non-uniformly whereas the TSV site design methodology places TSVs uniformly. Two algorithms developed for TSV assignment for the TSV site

design methodology are also presented. Simulation results compare 2D and 3D ICs, and show various trends found in gate-level 3D ICs.

- In Chapter 5, a design methodology and algorithms enabling block-level 3D IC design are developed and presented. Since the proposed design methodology requires the generation of routing topologies and a manipulation of whitespace, a 2D rectilinear Steiner minimum tree (RSMT)-based 3D rectilinear Steiner tree (RST) construction algorithm is developed and a sequential whitespace manipulation algorithm is proposed. Simulation results compare 2D and 3D floorplans, signal TSV planners, and various TSV insertion algorithms (single TSV insertion, 3D minimum spanning tree (MST)-based multiple TSV insertion, and 3D RST-based multiple TSV insertion).
- In Chapter 6, the impact of TSVs on the quality of 3D ICs is investigated. Since TSVs occupy silicon area, TSV insertion has a non-negligible impact on the area and the wirelength of 3D ICs. In addition, TSVs have capacitances that strongly influence the timing and the power of 3D ICs. Process technology has similar effects on the quality of ICs. In this chapter, therefore, 3D IC layouts are generated with various process and TSV technologies, and the impact of TSVs on the quality of 3D ICs is thoroughly studied.
- In Chapter 7, topography variation in 3D ICs is studied. Since TSVs are large and a TSV is connected to a metal landing pad, the landing pad is much larger than the wires. Unlike the large metal segments used in I/O cells, which are usually placed on the chip boundary, the TSV landing pads are placed inside the core area. Therefore, these large landing pads could cause serious topography variation in 3D ICs. Thus, to reduce topography variation, the topography variation caused by TSV landing pads is investigated, and a physical design technique is applied.
- In Chapter 8, the research presented in this dissertation is summarized.

CHAPTER 2

THROUGH-SILICON-VIA-AWARE INTERCONNECT PREDICTION OF MULTI-GRANULARITY THREE-DIMENSIONAL INTEGRATED CIRCUITS

Technology advances have pushed functional integration to such a high level that the interconnect and package represent real barriers to further progress. While significant research effort has been expended on several different technology fronts, three-dimensional (3D) integration is now emerging as a leading contender in the challenge of meeting performance, power, cost, and size demands through this decade and beyond. The 3D integrated circuit is an emergent technology that vertically stacks multiple dies with a die-to-die interconnect so called a through-silicon via (TSV). The TSV provides the possibility of arranging digital functional unit blocks across multiple dies at a very fine level of granularity. This results in a decrease in the overall wire length, which translates into less wire delay and less power. Advances in 3D integration and packaging are undoubtedly gaining momentum and have become of critical interest to the semiconductor community.

The advantage of shorter wirelength mainly originates from the usage of TSVs. However, TSVs do have its negative impact. For example, TSVs consume silicon area as shown in Figure 5, and the additional area for TSVs increases total chip area. Moreover, TSVs act as obstacles during placement and routing. Depending on the types of TSVs, via-first TSVs occupy the device layer, via-last TSVs occupy both the device and metal layers. Therefore, one may need to increase chip area to address the placement and routing congestion caused by TSV insertion. These factors are primarily due to the non-negligible size of the TSVs (1 to $10\mu\text{m}$ diameter typically). Therefore, their impact on area, power, and delay are indeed significant. However, most existing work related to TSVs, especially in the field of interconnect prediction tend to ignore the TSV size impact on the overall silicon footprint area of 3D ICs.

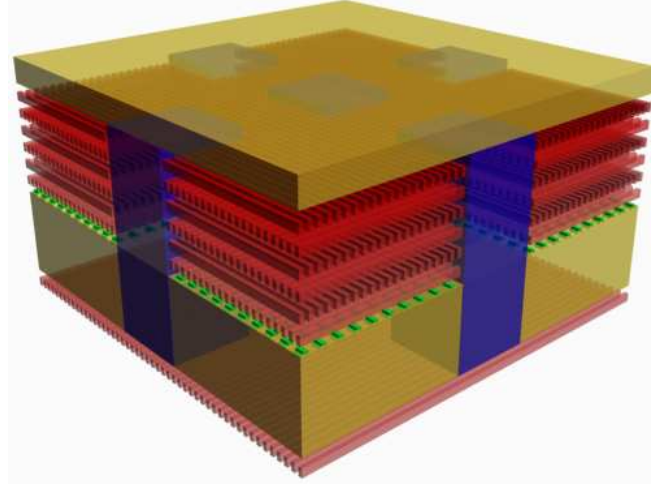


Figure 5. Via-last TSVs, where each TSV is surrounded by neighboring TSVs and wires. A typical size of TSV is much larger than that of global wires.

2.1 Related Work

After the successful prediction of Davis’ 2D wirelength distribution model [19], a few works have extended it to 3D wirelength distribution model [9, 14, 20]. While other work assumes one vertical pitch is same as one gate pitch, the authors of [9, 20] introduced a new parameter r , which is the strata-to-gate-pitch ratio. The strata pitch varies in a wide range depending on manufacturing technology such as die thinning, TSV materials, microfluidic channels for cooling, and so on. Since the gate placers should use less number of TSVs as the strata pitch goes up, the inclusion of r has a significant importance. However, [9, 20] do not provide closed-form formulas, so the computation time is very long.

[21] extended Davis’ 2D wirelength distribution model by introducing a new parameter p_{gates} , which is the percentage of die area occupied by logic gates. This is to explain the impact of whitespace existing in the placement layer. The authors show that the impact of p_{gates} on wirelength could be as large as 10% of the total wirelength.

While many studies have been done on 3D wirelength distribution, the impact of TSV size has not been mentioned in any of them. As a simple example, a $10\mu\text{m} \times 10\mu\text{m}$ signal TSV is comparable to about 50 gates in terms of area in 45nm technology. If one million TSVs of this size are used, the TSVs occupy area of 50 million gates, which is prohibitive.

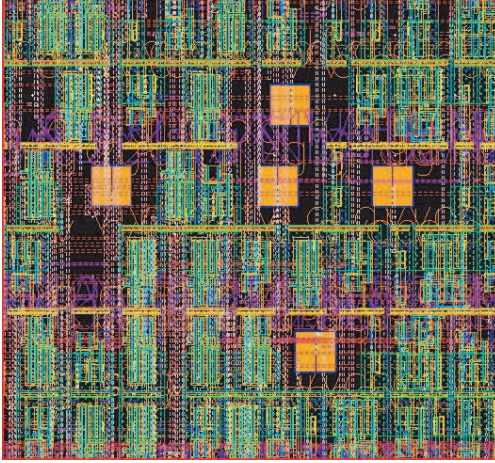


Figure 6. TSVs placed (orange square, shown in Cadence Virtuoso).

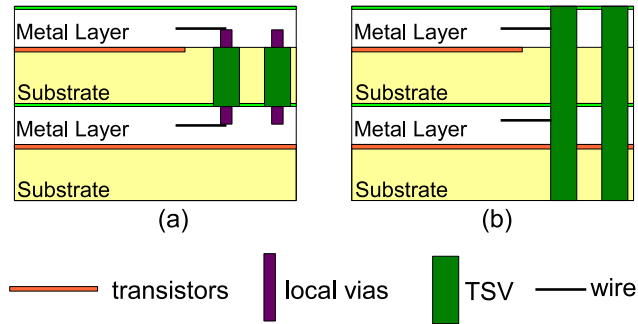


Figure 7. Two types of TSVs. (a) via-first (b) via-last

Therefore, TSV size and count should be considered in 3D ICs.

2.2 Preliminaries

The two most popular ways to fabricate TSVs are “via-first” and “via-last” processes, depending on when the via is implemented in the production process [22] (see Figure 7). Via-last TSVs are realized once the CMOS devices are completed and after the grinding and thinning process for wafer thinning. Via-last TSVs occupy all three layers: bulk, device, and metal layers, thereby becoming serious layout obstacles (see Figure 6). Via-first TSVs are implemented on the wafer prior to any production process, even before CMOS device fabrication. Via-first, however, is technically more challenging. Via-first TSVs are smaller compared to via-last, and TSVs occupy only two layers: bulk and device. This causes less interference with other layout objects.

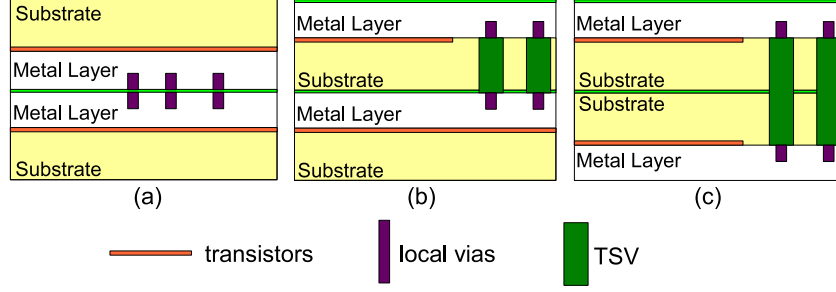


Figure 8. Three types of bonding. (a) F2F (Face-to-Face) (b) F2B (Face-to-Back) (c) B2B (Back-to-Back)

Figure 8 shows three different bonding styles. Face-to-face (F2F) uses local vias or relatively small TSVs to connect to other dies. On the other hand, face-to-back (F2B) and back-to-back (B2B) bonding do need TSVs to maintain TSV aspect ratio [23]. As TSV size becomes bigger, more silicon area is used for the same number of TSVs.

3D stacking can be done at three different levels of granularity: gate-level, block-level, and chip-level. The gate-level stacking allows individual gates to be placed in any die in the 3D stack, whereas the block-level stacking requires that all gates in the same block stay together in the same die. However, each block can be placed in any die in the stack. The last chip-level stacking simply stacks entire 2D dies without any inter-die optimization. In terms of the number of TSVs required, the gate-level stacking contains the highest TSV count, whereas the chip-level stacking requires the lowest TSV count.

2.3 TSV-Aware 3D Wirelength Distribution Model

2.3.1 TSV-Aware Chip Area Model

From this section, it is assumed that F2B bonding is applied to all dies. Other bonding styles, however, can be modeled in a similar way.

A TSV is inside a TSV cell which has some whitespace around the TSV depending on design rules (see Figure 6). It comes from the minimum distance between two adjacent TSVs, between a TSV and an adjacent metal wire, or between a TSV and a transistor.

Table 3 shows the notations used in our modeling. Assuming that TSVs are evenly

distributed between any two dies, the following equations hold:

$$A_{TSV,S} = N_{TSV,S} \cdot A_{TSVcell,S}$$

$$A_{TSV,PG} = N_{TSV,PG} \cdot A_{TSVcell,PG}$$

$$A_{3D} = A_{2D} + (N_{DIE} - 1) \cdot (A_{TSV,S} + A_{TSV,PG}) \quad (1)$$

$$A_{3DFP} = \frac{A_{3D}}{N_{DIE}}$$

$$\frac{A_{3D}}{A_{2D}} = 1 + (N_{DIE} - 1) \cdot \left(\frac{A_{TSV,S} + A_{TSV,PG}}{A_{2D}} \right) \quad (2)$$

$$\frac{A_{3DFP}}{A_{2DFP}} = \frac{1}{N_{DIE}} + \left(1 - \frac{1}{N_{DIE}} \right) \cdot \left(\frac{A_{TSV,S} + A_{TSV,PG}}{A_{2D}} \right) \quad (3)$$

Equation (31) shows the total silicon area of the 3D chip. Since F2B bonding is used, the bottommost die does not have TSVs. The additional silicon area, therefore, is the TSV area between two dies multiplied not by N_{DIE} but by $N_{DIE} - 1$.

A_{3D}/A_{2D} is 1, and A_{3DFP}/A_{2DFP} is $1/N_{DIE}$ if TSV size is zero. In this case, the additional silicon area becomes zero because $A_{TSVcell,S}$ and $A_{TSVcell,PG}$ are zero. However, the silicon area and footprint area ratios of 3D to 2D are strongly related to the occupancy rate of signal and P/G TSVs as shown in equation (32) and (3). Therefore, TSV size impact should be considered during wirelength prediction.

2.3.2 New 3D Design Parameters

Before deriving the new wirelength distribution model, a few parameters are introduced to explain various phenomena caused by TSV insertion in 3D.

- $P_{TSV,place}$: There are usually whitespaces in ICs [24] thus existing whitespaces can be used for TSV insertion. Some whitespaces, however, cannot be used because they are used for decap and minimization of congestion, the whitespaces are too small for TSV insertion, or they are far away from appropriate TSV locations. In these cases, silicon area needs to be increased for TSV insertion. The increased area is formulated

Table 3. Notations

$N_{TSV,S}$	average number of signal TSVs between two dies
$N_{TSV,PG}$	average number of P/G TSVs between two dies
$A_{TSVcell,S}$	area of a signal TSV cell
$A_{TSVcell,PG}$	area of a P/G TSV cell
A_{2D}	silicon area of a 2D chip
A_{3D}	total silicon area of a 3D chip
N_{DIE}	# dies
A_{2DFP}	footprint area of a 2D chip
A_{3DFP}	footprint area of a 3D chip
N_{gates}	total # gates
N_S	average # gates in a die
p_{gates}	the percentage of die area occupied by logic gates [21]
r	die-to-gate-pitch ratio in Figure 8 [9]
$M_S[l]$	# gate pairs separated by l gate pitches in a die
$M_z[v]$	# die pairs separated by v vertical pitches
$M_{tz}[l, v]$	# gate pairs separated by v vertical pitches and total l gate pitches
$M_t[l]$	# gate pairs separated by l gate pitches in 3D
$I_{exp}[l]$	# interconnects between two gate pairs separated by l gate pitches
Γ	normalization constant
$i(l)$	normalized wirelength distribution without TSV size
$i^*(l)$	normalized wirelength distribution with TSV size
$N_h(v)$	the number of wires whose vertical length is v

as

$$\Delta A_{TSV,place} = P_{TSV,place} \cdot A_{TSV} \quad (4)$$

$$P_{TSV,place} \geq 0 \quad (5)$$

where A_{TSV} is the total area of inserted TSVs. If $P_{TSV,place}$ is 0, TSVs can be inserted into the existing whitespace of the chip so that no additional silicon area is needed. If $P_{TSV,place}$ is 1, on the other hand, the chip area should be increased whenever TSVs are inserted because the existing whitespace cannot be used for TSV insertion. $P_{TSV,place}$ can be greater than 1 because insertion of a TSV cell may need a row creation if the design is based on standard cell libraries.

- $P_{TSV,route}$: For via-first fabrication (Figure 7(a)), routing congestion is mainly caused

by connections between metal wires and TSVs. For via-last fabrication (Figure 7(b)), routing congestion is mainly caused by inserted TSVs which make wires bypass the TSVs. This parameter is for the explanation of the different degree of routing congestion caused by various types of TSVs and bonding styles, and circuit characteristics such as # nets, # gates, and so on. The increased area is formulated as

$$\Delta A_{TSV,route} = P_{TSV,route} \cdot A_{TSV} \quad (6)$$

$$P_{TSV,route} \geq 0 \quad (7)$$

where A_{TSV} is the total area of inserted TSVs. If $P_{TSV,route}$ is 0, no routing congestion is caused by TSV insertion, which happens when there are already enough space for connection between metal wires and TSVs. If $P_{TSV,route}$ is greater than 0, on the other hand, some wires bypassing TSVs cause congestion around the TSVs. In this case, whitespace should be inserted to resolve the congestion.

Then, the total silicon area in equation (31) becomes

$$A_{3D} = A_{2D} + \Delta A_{TSV,place} + \Delta A_{TSV,route} \quad (8)$$

- B (Granularity parameter) : Placement can be done at gate-level or block-level. A specific granularity of block size can also be chosen for block-level placement. This parameter explains how big blocks are used for 3D placement and is defined as the average number of blocks in a die. Therefore, gate-level placement is done when B is equal to N_{gates}/N_{DIE} and the most coarse block-level placement is done when B is 1.

$$1 \leq B \leq \frac{N_{gates}}{N_{DIE}} \quad (9)$$

2.3.3 Gate-level 3D Wirelength Distribution

This section shows the new wirelength distribution considering TSV size at gate-level (see Table 3 for the notations). The normalized wirelength distribution without consideration of

TSV size in [9, 20] is as follows:

$$M_{tz}[l, v] = M_z[v]M_S[l - vr] + N_S(N_{\text{DIE}} - v)\delta[l - vr] \quad (10)$$

$$M_t[l] = \sum_{v=0}^{N_{\text{DIE}}-1} M_{tz}[l, v] \quad (11)$$

$$i(l) = \Gamma \cdot M_t[l] \cdot I_{exp}[l] \quad (12)$$

The first term in the right-hand side of equation (10) becomes zero for wires whose horizontal length is zero (call these *PV* wires). On the other hand, the second term becomes zero for wires whose horizontal length is nonzero (call these *NPV* wires).

PV wires are not affected by TSV insertion because their horizontal wirelength is zero while the horizontal wirelength of *NPV* wires are affected by TSV insertion. Therefore Equation (12) is rewritten as follows:

$$i(l) = \Gamma \cdot M_t[l] \cdot I_{exp}[l] = i_h(l) + i_v(l) \quad (13)$$

$$i_h(l) = \Gamma \cdot I_{exp}[l] \cdot \sum_{v=0}^{N_{\text{DIE}}-1} M_z[v]M_S[l - vr] \quad (14)$$

$$i_v(l) = \Gamma \cdot I_{exp}[l] \cdot \sum_{v=0}^{N_{\text{DIE}}-1} N_S(N_{\text{DIE}} - v)\delta[l - vr] \quad (15)$$

where $i_h(l)$ consists of *NPV* wires and $i_v(l)$ consists of *PV* wires. Only $i_h(l)$ is modified because TSV size affects only the horizontal wirelength.

$i_h(l)$ in Equation (14) can be rewritten as follows:

$$i_h(l) = \sum_{v=0}^{N_{\text{DIE}}-1} i_h(v, l) \quad (16)$$

$$i_h(v, l) = \Gamma \cdot I_{exp}[l] \cdot M_z[v]M_S[l - vr] \quad (17)$$

$$N_h(v) = \sum_{l=1}^{2\sqrt{N_S}} i_h(v, l) \quad (18)$$

where $i_h(v, l)$ is the wirelength distribution of wires whose total length is l gate pitches and vertical length is v vertical pitches. $N_h(v)$ is the number of *NPV* wires whose vertical length is v vertical pitches. This number should be conserved for each v .

Then, the new wirelength distribution is derived by re-normalization as follows:

$$i_h^*(v, l) = \Gamma(v)^* \cdot I_{exp}^*[l] \cdot M_z[v] M_S^*[l - vr] \quad (19)$$

$$\Gamma(v)^* = \frac{N_h(v)}{\sum_{l=1}^2 \sqrt{N_S^*} I_{exp}^*[l] \cdot M_z[v] M_S^*[l - vr]} \quad (20)$$

$$N_S^* = A_{3D}/N_{DIE} \quad (21)$$

where $\Gamma(v)^*$ is the re-normalization constant for NPV wires whose vertical length is v vertical pitches, $I_{exp}^*[l]$ is the modified expected number of interconnects connecting two gate socket pairs at a distance of l , and $M_S^*[l]$ is the modified total number of gate socket pairs at a distance of l . As seen in the above equations, $i_h^*(v, l)$ was re-normalized separately.

Then the new distribution becomes as follows.

$$i_h^*(l) = \sum_{v=0}^{N_{DIE}-1} i_h^*(v, l) \quad (22)$$

$$i^*(l) = i_h^*(l) + i_v(l) \quad (23)$$

The modified $M_S^*[l]$ is as follows:

$$M_S^*[l] = M_S[l] - OVR[l] \quad (24)$$

where $OVR[l]$ is calculated by computing the number of gate pairs which are l gate pitches away. One of the gates in the pair should be inside a TSV cell.

$I_{exp}^*(l)$ is computed as follows.

$$\begin{aligned} N_A^* &= 1 \\ N_B^*[l] &= N_B[l] \cdot (1 - R_{tg}) \cdot p_{gates} \\ N_C^*[l] &= N_C[l] \cdot (1 - R_{tg}) \cdot p_{gates} \\ I_{exp}^*[l] &= \frac{p_{gates} \cdot \alpha k}{N_A^* N_C^*[l]} \left[\frac{(N_A^* + N_B^*[l])^p + (N_B^*[l] + N_C^*[l])^p}{-(N_B^*[l])^p - (N_A^* + N_B^*[l] + N_C^*[l])^p} \right] \\ R_{tg} &= \frac{L_{TSV}}{\sqrt{A_{3D}/N_{TSV}}} \end{aligned}$$

- | |
|---|
| 1: Compute average intra-block wirelength
2: Compute average inter-block wirelength
3: Compute # TSVs
4: Compute average intra-block wirelength with TSV size
5: Compute average inter-block wirelength with TSV size |
|---|

Figure 9. Derivation of block-level 3D wirelength distribution

where L_{TSV} is the width of a TSV cell, and N_{TSV} is the total number of TSVs. In the model, the gate pitch, which is same as L_{gate} , is fixed. Therefore, p_{gates} is defined as follows:

$$p_{gates} = \frac{N_{gates}}{A_{2D}/L_{gate}^2} \quad (25)$$

2.3.4 Block-level 3D Wirelength Distribution

The modeling of block-level 3D wirelength distribution is done hierarchically. The derivation flow is shown in Figure 9.

First, intra-block wirelength without TSV size is computed by 2D wirelength distribution. In order to compute intra-block wirelength, equations in [21] are used. The total number of interconnects in a block is calculated as follows:

$$I_{B,intra} = \alpha \cdot k \cdot N_{Bg} \cdot (1 - N_{Bg}^{p-1}) \quad (26)$$

where α , k and p are Rent's constants [25], and N_{Bg} is the number of gates in a block.

Next, inter-block wirelength without TSV size is computed by 3D wirelength distribution. In this case, however, a block is treated as a gate and Equation (12) is applied. The total number of inter-block connections is calculated as follows:

$$I_{total} = \alpha \cdot k \cdot N_{gates} \cdot (1 - N_{gates}^{p-1}) \quad (27)$$

$$I_{B,inter} = I_{total} - I_{B,intra} \cdot N_{DIE} \cdot B \quad (28)$$

where I_{total} is the total number of interconnects in the circuit and B is the granularity parameter.

For the computation of the number of TSVs, it is assumed that an inter-block connection exists between two gates separated by distance D_B . D_B is defined as follows:

$$D_B = n_H \cdot L_H + n_V \cdot r \quad (29)$$

where L_H is the average distance between two adjacent blocks in a die, and n_H and n_V are integers greater than or equal to zero. Then the total number of TSVs is computed in a similar way shown in [9].

Then, the number of TSVs in a block is computed as follows:

$$N_{B,inter,TSV} = \frac{N_{B,TSV}}{(N_{DIE} - 1) \cdot B} \quad (30)$$

where $N_{B,TSV}$ is the total number of TSVs obtained during the computation of the number of TSVs. TSVs inserted into a block increase the block area. This area affects the wirelength of intra-block wires. The intra-block wirelength with TSV size is computed in a similar way shown in Equation (19).

The increased block area also affects the wirelength of inter-block *NPV* wires because the distance between two gates from two different blocks is increased. On the other hand, it is assumed that *PV* wires in block-level distribution are not affected by the increased block area for simplification. The computation of inter-block wirelength with TSV size is done by Equation (19).

2.4 Validation

2.4.1 Validation of TSV Count

The authors of [26] designed 3D chips by folding 2D designs for various benchmarks. This design scheme is similar as block-level placement thus their TSV counts in [26] are compared to our block-level prediction.

Table 4 shows the comparison of TSV count for all the circuits reported in [26]. The table shows that our predictions match well with the reported numbers in most cases, although absolute difference in some cases is up to 30%.

Table 4. Validation of our prediction on TSV count for block-level placement.

circuit	[26] Folding-2	ours ($B=1$)	Dif. (%)	[26] Folding-4	ours ($B=4$)	Dif. (%)
ibm01	1,671	1,595	4.55	2,476	3,852	-55.57
ibm03	4,125	2,487	39.71	5,909	6,006	-1.64
ibm04	2,940	2,850	3.06	6,388	6,883	-7.75
ibm06	4,116	3,285	20.19	9,077	7,933	12.60
ibm07	5,932	4,233	28.64	8,755	10,222	16.76
ibm08	5,801	4,638	20.05	10,181	11,199	-10.00
ibm09	4,540	4,690	-3.30	8,257	11,326	-37.17
ibm13	7,696	6,594	14.32	13,071	15,923	-21.82
ibm15	15,128	10,845	28.31	23,662	26,187	-10.67
ibm18	12,077	13,425	-11.16	28,287	32,415	-14.59
	Absolute Dif.		17.33	Absolute Dif.		18.86

Table 5. Validation of our prediction on wirelength. $N_{DIE} = 3$, $P_{TSV,place} = 1$ and $P_{TSV,route} = 0$.

				Wirelength (μm)				
circuit	# gates	# nets	# TSVs	3D Design	Prediction (w/o TSV size)	Dif.	Prediction (with TSV size)	Dif.
Ind 1	11,295	11,839	688	315,792	276,465	-12.45%	300,540	-4.83%
Ind 2	29,706	29,979	1,217	569,482	452,222	-20.59%	488,465	-14.22%
wb_conmax	62,028	63,158	719	1,108,165	953,519	-13.96%	994,221	-10.28%
Ind 3	260,579	262,357	1,799	6,991,301	6,320,040	-9.60%	6,504,680	-6.96%
netcard	651,674	653,155	2,261	23,303,968	21,168,400	-9.16%	21,560,700	-7.48%

2.4.2 3D Circuit Design Scheme

Figure 10 shows our 3D circuit design scheme for validation of 3D wirelength distribution. First, HDL source files are synthesized with Synopsys Design Compiler [27]. Then N-way partitioning is performed for N-die designs here N is the number of dies ($=N_{DIE}$). The area balancing factor used is 0.05 (5%). Since TSV cells will be inserted to all dies except the bottommost die and inserting TSV cells increases die area, dies are sorted in the order of die area before TSV insertion so that the largest die is laid at the bottommost location. After inserting TSV cells, placement is performed by Cadence SoC Encounter [28] for the topmost die (3D_1 in Figure 10) and extract TSV cell locations. Since the locations of these cells affect the placement of the next die (3D_2), the locations of the cells are fed into SoC Encounter during the placement of 3D_2. Placement for the remaining dies is done in a similar way.

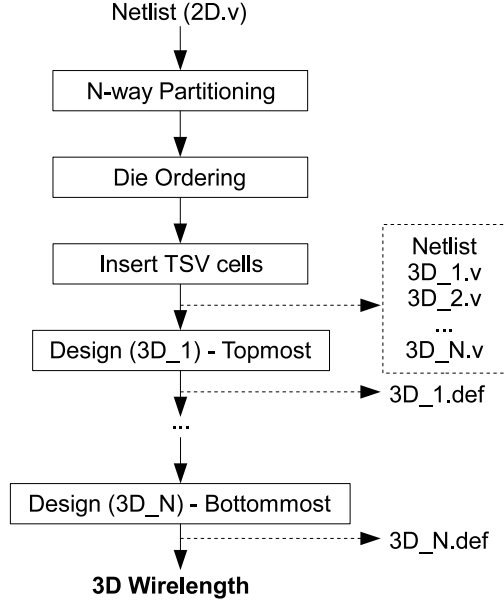


Figure 10. 3D Circuit Design Scheme

Then routing (both global and detailed) is done for each placement result (3D_#.def). Cadence SoC Encounter is used for routing.

Figure 11 shows the snapshot of TSV cells in Cadence SoC Encounter. Yellow points are TSV cells and black area is actually filled with standard cells which are not shown. Figure 12 shows all the connections to TSV cells.

2.4.3 Validation of Wirelength Prediction

Three industrial circuits (Ind 1, Ind 2 and Ind 3) and two IWLS'05 benchmark circuits (wb_conmax and netcard) [29] are used for validation of our wirelength prediction.

Table 5 shows the comparison of total wirelength for each circuit. As seen in the table, both predictions underestimate for the three circuits but prediction considering TSV size is more accurate. Since this 3D design method does not optimize gate placement globally, the wirelength of circuits globally optimized in 3D with the same number of TSVs will be shorter than the wirelength in Table 5. Then the prediction will match the wirelength more closely.

In order to make the prediction even more accurate, it is necessary to determine the

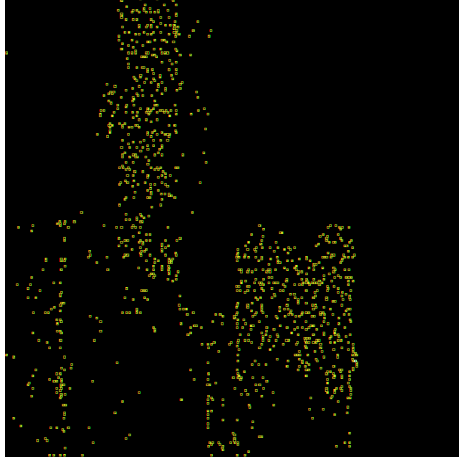


Figure 11. Snapshot of TSVs inserted in the topmost die of the circuit “Ind 3” (see Table 5) in Cadence SoC Encounter. There are 1186 TSVs (yellow) and 80483 standard cells (black). Die area is $592\mu m \times 592\mu m$.

Table 6. Impact of TSV size consideration on wirelength. $N_{DIE} : 4$, $p_{gates} : 1$, $L_{gate} : 1.37\mu m$ and $L_{TSV} : 1.37\mu m$. $B = N_{gates}/N_{DIE}$ (gate-level).

r	N_{gates}	# TSVs	2D WL	TSV size consideration	3D WL	Δ WL
5	1M	0.66M	17.23	no	11.23	-34.82%
				yes	14.35	-20.07%
	100M	75.2M	53.96	no	29.82	-44.74%
				yes	40.00	-25.87%
30	1M	0.17M	17.23	no	13.37	-22.40%
				yes	14.82	-13.99%
	100M	24.3M	53.96	no	30.37	-43.72%
				yes	34.51	-36.05%

parameters such as $P_{TSV,place}$ and $P_{TSV,route}$ more carefully. In Table 5, fixed $P_{TSV,place}$ and $P_{TSV,route}$ are used regardless of circuit characteristics to show how our prediction behaves.

2.5 Impact Study

Table 6 shows the impact of TSV size on wirelength. As the table shows, the wirelength difference is between 10% and 20%. If r is small, 3D placers tend to use more TSVs and the difference becomes greater. This is because the TSVs occupy larger space so that silicon area increase by TSV insertion affects the wirelength significantly. The difference also becomes more noticeable if the TSV size is relatively bigger than the gate size.

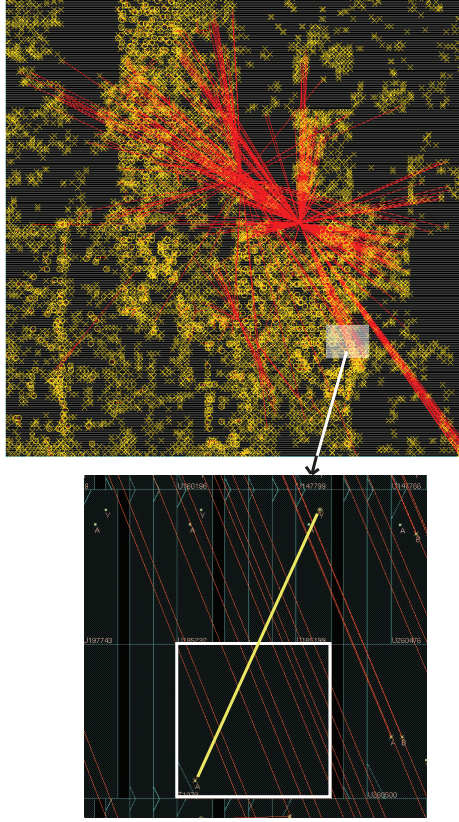


Figure 12. Snapshot of connections to TSVs in the topmost die of the circuit “Ind 3” in Cadence SoC Encounter. There are 1186 connections to TSVs among 82167 nets. Die area is $592\mu\text{m} \times 592\mu\text{m}$. The white square is a TSV cell and thick yellow line shows the connection between the TSV cell and an inverter cell.

The Rent’s constants in the experiments are $\alpha = 0.75$, $k = 4$ and $p = 0.75$. The parameters used for this study are as follows (see Table 3). $N_{\text{gates}} = 40M$. L_{gate} , which is the physical gate width, is $1.37\mu\text{m}$. The variable parameters are as follows (if not specified in each case): 2D die size = 100mm^2 , $L_{\text{TSV}} = 1.37\mu\text{m}$, $P_{\text{TSV},\text{place}} = 1$, $P_{\text{TSV},\text{route}} = 0$, $r = 30$ and $N_{\text{DIE}} = 2$. Lastly, gate-level stacking option, where $B = N_{\text{gates}}/N_{\text{DIE}}$, is used.

2.5.1 Impact of TSV Size and Design Parameters

- TSV size (Figure 13) : As TSV size increases, silicon area and footprint area increase, and so does wirelength. 3D WL becomes bigger than 2D WL if TSV size continues to go up, which means that it is not possible to benefit from 3D with respect to WL. **In short, silicon area, footprint area and WL increase as TSV size increases.**

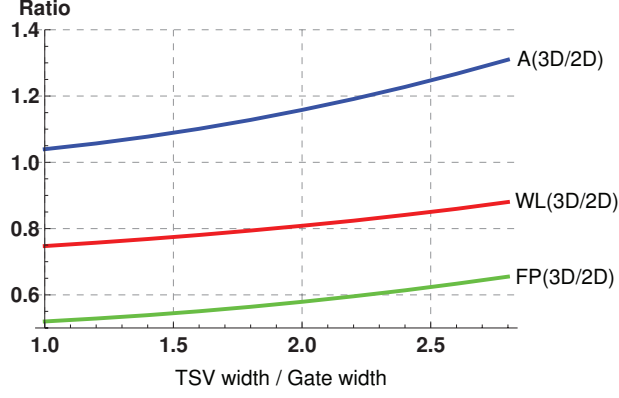


Figure 13. Impact of TSV size on silicon area (A), footprint area (FP) and wirelength (WL). $r : 100$

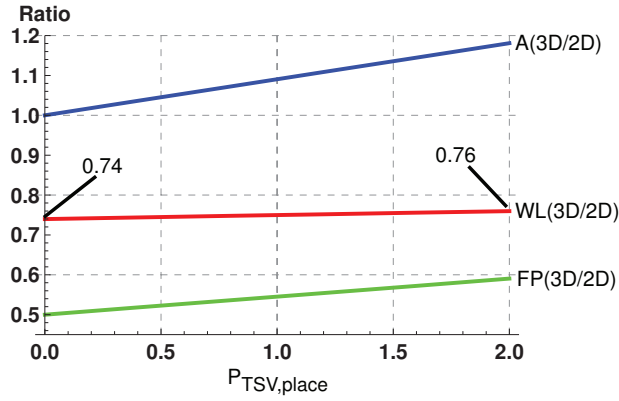


Figure 14. Impact of $P_{TSV,place}$ on silicon area (A), footprint area (FP) and wirelength (WL).

- $P_{TSV,place}$ (Figure 14) : 3D silicon area increases as $P_{TSV,place}$ goes up. Therefore, gates are spread out so that WL slightly increases. Since the silicon area increase strongly depends on TSV size as well as TSV count, the three ratios will become bigger if TSV size or TSV count increases. Moreover, even though the WL increase in this figure is small, this parameter should be kept as small as possible to save the cost for silicon area, i.e., better placement tool is needed. **In short, silicon area, footprint area and WL increase as $P_{TSV,place}$ increases.**
- $P_{TSV,route}$ (Figure 15) : This parameter affects the three metrics in the same way as $P_{TSV,place}$. However, the range of $P_{TSV,route}$ is larger than $P_{TSV,place}$ because of via-last fabrication. If a circuit is seriously congested, via-last TSVs will cause many wires to overlap with each other, thereby requiring $P_{TSV,route}$ to be more than 1 or 2. Figure 15

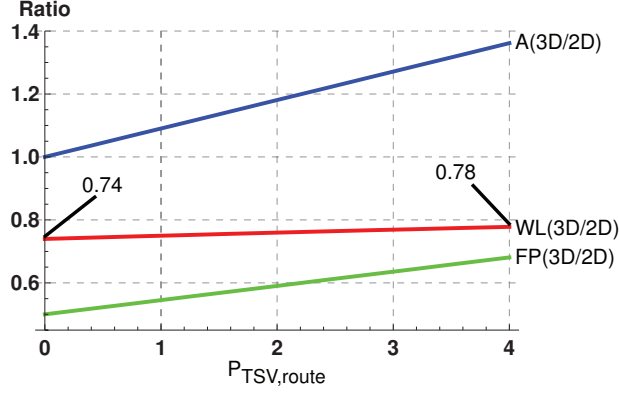


Figure 15. Impact of $P_{TSV,route}$ on silicon area (A), footprint area (FP) and wirelength (WL).

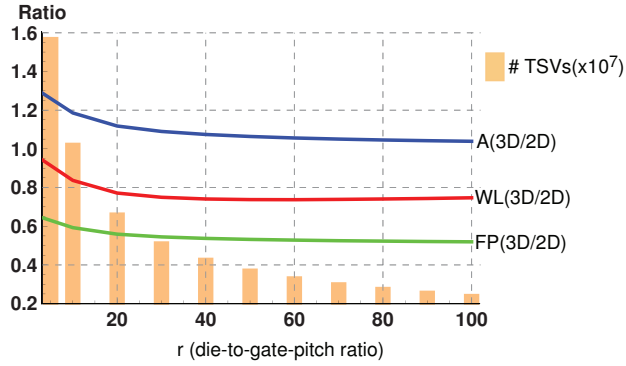


Figure 16. Impact of r on silicon area (A), footprint area (FP) and wirelength (WL).

confirms a similar impact trend as $P_{TSV,place}$. **In short, silicon area, footprint area and WL increase as $P_{TSV,route}$ increases.**

- r (Figure 16) : Bigger r , which means taller die, means placers use less number of vertical connections. As the figure shows, the number of TSVs decrease as r increases. The silicon area, footprint area and WL also decrease because less number of TSVs are used. However, note that the number of TSVs cannot be decreased below the min-cut size in real circuits. **In short, silicon area, footprint area and WL decrease as r increases.**
- N_{DIE} (Figure 17) : More and more TSVs are used when the number of dies increases. This increases silicon area, but WL decreases because of more vertical connections. The WL decrease saturates at some point. **In short, silicon area increases but**

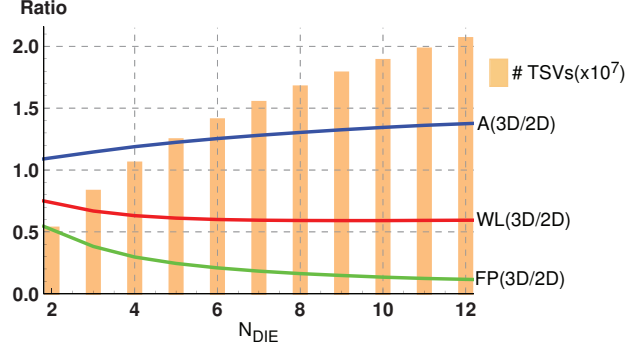


Figure 17. Impact of N_{DIE} on silicon area (A), footprint area (FP) and wirelength (WL).

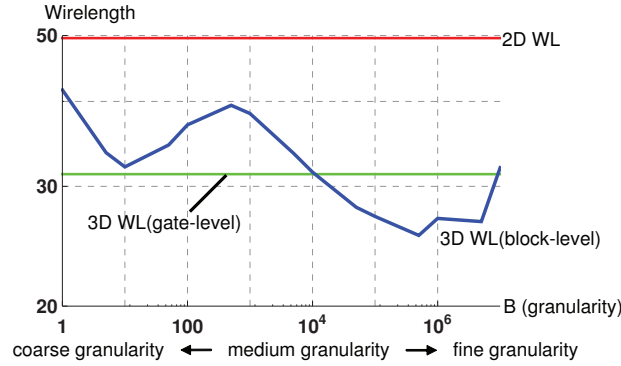


Figure 18. Impact of B on wirelength (WL). $N_{DIE} : 4$.

footprint area and WL decrease as N_{DIE} increases.

- B (Figure 18, 19 and 20) : As expected, when B is 1, only one big block exist in each die (coarse granularity). So, the silicon area increase is small, but WL decrease is also small. As B goes up, silicon area and footprint area generally increase, but WL fluctuates. Wirelength reaches the minimum usually at fine granularity at which one block has about 20 to 100 gates. Area ratio reaches the minimum when B is 1. **In short, silicon area and footprint area ratios increase but saturate at some point. On the other hand, WL fluctuates and reaches the minimum at fine granularity.**

2.5.2 Case Study

A case study is shown in this section to demonstrate how to use our model for early decision making for 3D ICs. The technology parameters used are as follows: number of gates is 10M under 90nm. TSV is via-last and its size is $2 \times 2\mu m$. Die height ratio r is 20, and the 2D die

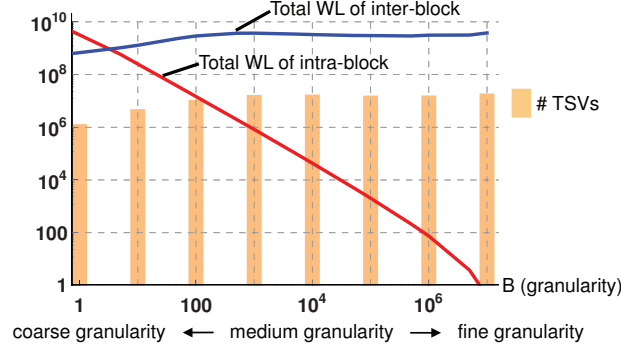


Figure 19. Impact of B on total wirelength of intra-block and inter-block, and TSV count. $N_{DIE} : 4$.

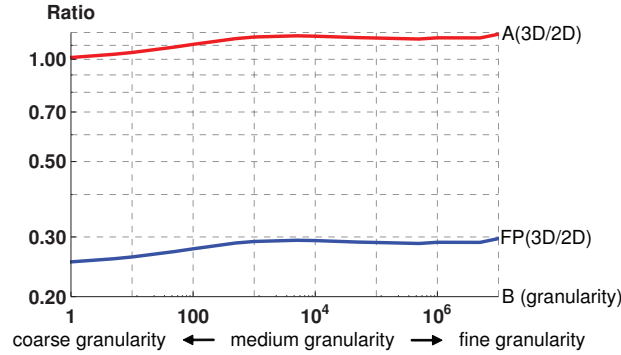


Figure 20. Impact of B on silicon area (A) and footprint area (FP). $N_{DIE} : 4$.

area is $78mm^2$. Lastly, $P_{TSV,place}$ is set to 1.0 and $P_{TSV,route}$ is set to 0.5.

These are the steps in the decision making: (1) For a circuit to be fabricated in 3D, the number of gates is calculated or predicted. (2) Fabrication technologies for the circuit and TSVs including bonding and TSV types are selected. (3) The circuit is simulated in 2D with existing tools to estimate how much decap is necessary, how serious the congestion is, how much the power consumption is, and so on. (4) Two additional parameters, $P_{TSV,place}$ and $P_{TSV,route}$, are estimated. (5) N_{DIE} and B are varied to estimate how many TSVs are used, how large the additional silicon area is needed and how much the wirelength is decreased. Table 7 shows the simulated values.

Based on this result, it is observed that 3D placement in “large” block-level stacking (coarse granularity) uses less number of TSVs, thereby achieving smaller silicon area increase while decreasing sufficient amount of wirelength. On the other hand, 3D placement

Table 7. Case study for early design exploration

N_{DIE}	B (Granularity)	# TSVs (mil)	$\frac{A_{3D}}{A_{2D}}$	Δ WL (%)
2	coarse	0.394	1.030	-19.63
	medium	1.164	1.089	-0.91
	fine	1.300	1.100	-21.52
	gate-level	1.453	1.112	-14.13
4	coarse	0.456	1.035	-23.37
	medium	2.362	1.181	-12.81
	fine	2.626	1.202	-32.84
	gate-level	3.005	1.231	-24.44
6	coarse	0.479	1.037	-24.10
	medium	3.073	1.236	-18.98
	fine	3.463	1.266	-35.60
	gate-level	4.017	1.308	-26.29

in “small” block-level stacking (finer granularity) uses many TSVs but decreases wirelength a lot. Our choice depends on what the most important factor is. If the yield of TSV fabrication is low so that TSV cost is high, coarse granularity 3D placement is the best option. If the TSV cost is low but die bonding cost is high, 2-die or 3-die stacking with fine granularity is the best choice. If TSV and die bonding costs are low and silicon area is not a concern, 6-die stacking with fine granularity is the best. This case study shows that medium granularity is worse than coarse granularity stacking with respect to TSV count, silicon area ratio, and wirelength. However, the trends vary depending on technologies, circuit size, and so on, as seen in Figure 18.

2.6 TSV-Aware Delay and Power Prediction Model for Buffered Interconnects in 3D ICs

Buffer insertion on TSV-based 3D interconnects is also non-trivial because buffers have non-trivial area overhead. In addition, both TSVs and buffers occupy device and M1 layers, so they cannot overlap with each other. Therefore, buffer model has to address these issues when used to predict the impact on delay and power of 3D interconnects. In this section, TSV-aware delay and power prediction models for buffered interconnects in 3D ICs are developed based on the TSV-aware 3D wirelength distribution model presented in

Table 8. Variables affecting TSV capacitance and their effects

Variable	Effects	
	Capacitance	Chip
TSV width (or diameter) ↑	MOS cap. ↑ MOM coupling cap. ↑	Die area ↑ Wirelength ↑
TSV height ↑	MOS cap. ↑	
Liner oxide thickness ↑	MOS cap. ↓	Die area ↑ Wirelength ↑

Table 9. Variation of TSV capacitance

Dimension (μm)			Capacitance (fF)	
Width	Height	Liner oxide thickness	MOS cap.	Coupling cap.
2.0	40.0	0.1	133.7	0.6
4.0	40.0	0.1	261.2	2.4
2.0	10.0	0.1	33.4	0.6
2.0	20.0	0.1	66.9	0.6
2.0	10.0	0.1	33.4	0.6
2.0	10.0	1.0	4.7	5.3

the previous sections.

2.6.1 TSV Resistance

TSV resistance consists of a material resistance ($= \rho \cdot \frac{l}{S}$) of a TSV itself and the contact resistance between a TSV and a landing pad at both ends of the TSV. The material resistance of a TSV is small in general because the cross-sectional area of a TSV is much bigger than that of a wire. For instance, assuming 1) TSV is made of tungsten, 2) TSV width is $2\mu m$, and 3) TSV height is $20\mu m$, the material resistance is $280m\Omega$ which is much smaller than the resistance of a very short wire. On the other hand, the contact resistance is strongly dependent on TSV manufacturing and die bonding technologies. In our simulation, 100Ω is used for the baseline TSV resistance, which is the sum of the material resistance and the contact resistance.

Table 10. Parameters and assumptions used in this paper

Parameter or assumption	Value
TSV shape	Square
TSV type	Via-first
Die-bonding type	Face-to-back
TSV width + liner oxide thickness + keep-off distance	$2.47\mu m$
TSV resistance	100Ω
Die-to-gate-pitch ratio (r) [9]	40
Device technology	$45nm$
Wire resistance of intermediate metal layers	$3.31\Omega/\mu m$
Wire capacitance of intermediate metal layers	$0.171fF/\mu m$
Output resistance of a $20\times$ buffer	305Ω
Input capacitance of a $1\times$ buffer	$1.55fF$
Buffer delay	$70ps$
Buffer switching energy	$6.65fJ$
Buffer switching power @ $1GHz$	$6.65\mu W$
Buffer switching activity	0.3
L_{FIX} (see Figure 24)	$350\mu m$
Cell switching energy (avg.)	$7.28fJ$
Cell switching power (avg.) @ $1GHz$	$7.28\mu W$
Cell switching activity	0.5
Rent's parameter α	0.75
Rent's parameter k	4.0
Rent's parameter p	0.75
Gate pitch	$1.37\mu m$
Output resistance of a $1\times$ buffer	R_{drv}
Input capacitance of a $1\times$ buffer	C_{drv}
Wire resistance per unit length	r_{wire}
Wire capacitance per unit length	c_{wire}
Buffer size	S_{buf}
Buffer delay	D_{buf}

2.6.2 TSV Capacitance

Assuming the bulk silicon around a TSV is DC-biased well as discussed in [30], TSV capacitance consists mainly of TSV(M)-Insulator(O)-Silicon(S) capacitance and TSV(M)-Insulator(O)-Wire(M) coupling capacitance as shown in Figure 21. The variables affecting these MOS and MOM TSV capacitances and their effects are shown in Table 8. As the table shows, TSV width and height are strongly related to TSV capacitance as well as the die area and the total wirelength in 3D ICs. For example, if the liner oxide thickness increases, TSV MOS cap decreases but the die area increases, and so does the total wirelength.

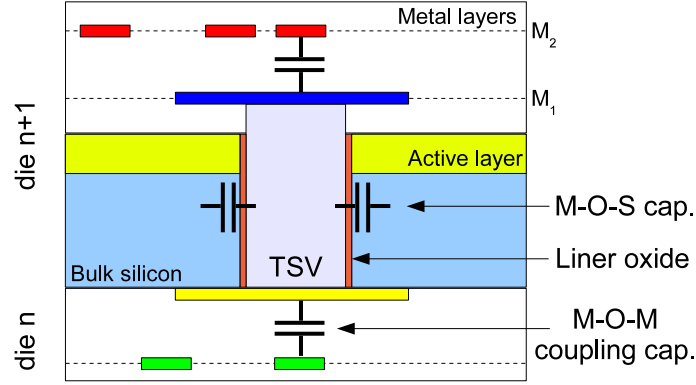


Figure 21. Via-first TSV and its capacitive components in face-to-back die bonding

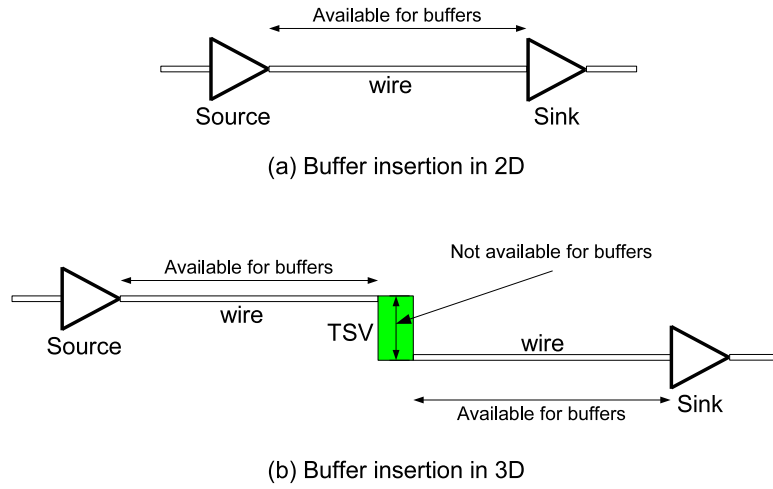


Figure 22. Buffer insertion in 2D and 3D ICs

Table 9 shows the capacitances of various combinations of TSV dimensions such as TSV width, height and the liner oxide thickness. These values are obtained by simulating the structures with Synopsys Raphael [31]. The assumptions for this simulation are shown in Table 10). As Table 9 shows, TSV capacitance varies in a wide range depending on TSV width and height, and the liner oxide thickness. Therefore, the TSV capacitance is varied from $5fF$ to $50fF$ in our simulation to cover the wide range of TSV capacitance.

2.6.3 Buffer Insertion Schemes

Theoretically a metal wire can be cut and split into two segments, and a buffer can be inserted into the cut point if there exists enough empty space to insert buffers along the

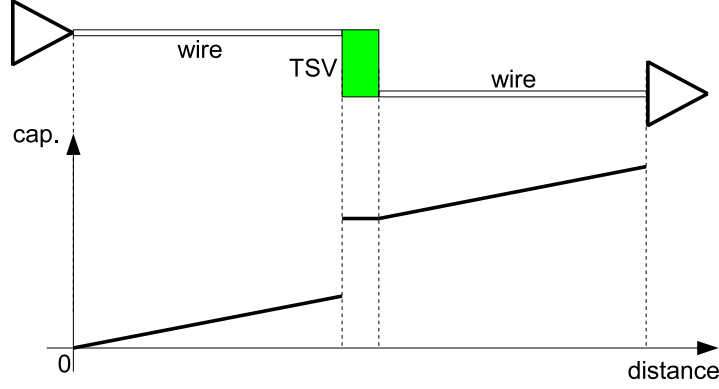


Figure 23. Distance-capacitance plot in a 3D wire

wire. On the other hand, buffers cannot be inserted inside a TSV. Buffers actually can be inserted into one end or both ends of a TSV. In other words, buffers for metal wires that have no TSVs can be inserted anywhere along the wires, whereas buffer insertion for 3D interconnects that contain TSVs has to avoid TSV obstacles. This is illustrated in Figure 22 and Figure 23.

In our delay prediction model, two buffer insertion schemes are used for 2D and 3D ICs to illustrate the impact of TSV RC on delay and power. Our buffering scheme is based on the consideration that buffer insertion cannot be too much detailed during wirelength prediction. The first scheme, **BIS1** (Buffer Insertion Scheme 1), is to insert a buffer at every fixed distance as shown in Figure 24(a). This applies to both 2D ICs and 3D ICs. In 3D ICs, a buffer is also inserted in front of a TSV to increase the driving strength for the TSV. The second scheme, **BIS2** (Buffer Insertion Scheme 2), is same as BIS1 but an additional buffer is inserted at the end of a TSV as shown in Figure 24(b) so that the TSV RC effect on delay is minimized. The distance-capacitance plots for BIS1 and BIS2 are shown in Figure 24.

Another assumption used in our simulation for buffer insertion is that a buffer is in fact a buffer chain. The first buffer in the buffer chain is a $1\times$ buffer, thus the input capacitance is minimized. The last buffer in the chain is a $20\times$ buffer, thus the output resistance becomes sufficiently small. The buffers between them are properly scaled based on the process

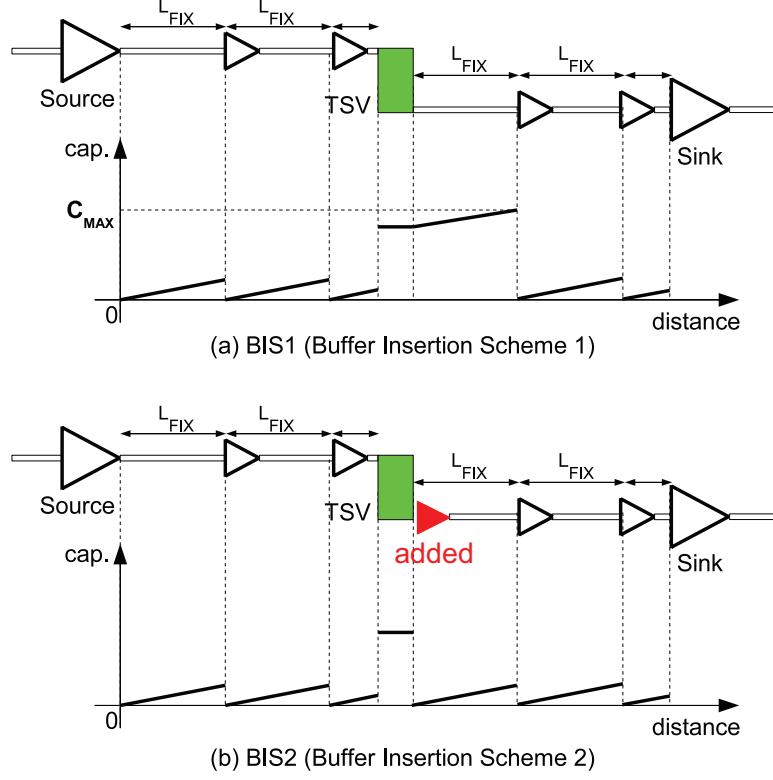


Figure 24. Two buffer insertion schemes used in this paper and their distance-capacitance plot

technology. The internal delay of the buffer chain is $70ps$.

2.6.4 Delay Computation

In this section, delay computation for 3D wires is explained briefly.

- 3D Wire Delay without TSV RC and without Buffer Insertion: When TSV RC is ignored, TSV height is added into wirelength to include the impact of TSVs on delay. Therefore, TSVs are considered as plain wires in this case. Then the delay of a wire whose length is $L(\mu m)$ is computed using Elmore delay as follows:

$$\begin{aligned}
 D_1(L) = & \frac{R_{drv}}{S_{buf}} \cdot c_{wire} \cdot L + \frac{1}{2} \cdot r_{wire} \cdot c_{wire} \cdot L^2 \\
 & + \left(\frac{R_{drv}}{S_{buf}} + r_{wire} \cdot L \right) \cdot C_{drv}
 \end{aligned} \tag{31}$$

where the definitions of the variables are shown in Table 10.

- 3D Wire Delay without TSV RC and with Buffer Insertion: If buffer insertion is considered, a 3D wire of length $L(\mu m)$ is split into segments of length L_{FIX} (Figure 24). The delay of a wire segment of length L_{FIX} is computed by $D_1(L_{FIX})$ in Equation (31). If the length of a 3D wire is not a multiple of L_{FIX} , the delay of the remaining length is also added. The delay of a 3D wire is then computed as follows:

$$D_2(L) = n_{buf} \cdot (D_1(L_{FIX}) + D_{buf}) + D_r \quad (32)$$

where n_{buf} is the number of buffers in the wire and D_r is the wire delay of the remaining length.

- 3D Wire Delay with TSV RC and with Buffer Insertion: If a wire is a *PV* wire whose horizontal length is negligible [32], the delay of the wire is computed as follows:

$$\begin{aligned} D_{PV} = & N_{TSV} \cdot \frac{R_{drv}}{S_{buf}} \cdot C_{TSV} + \frac{1}{2} \cdot N_{TSV}^2 \cdot R_{TSV} \cdot C_{TSV} \\ & + \left(\frac{R_{drv}}{S_{buf}} + N_{TSV} \cdot R_{TSV} \right) \cdot C_{drv} \end{aligned} \quad (33)$$

where N_{TSV} is the number of TSVs in the wire, R_{TSV} is the TSV resistance, and C_{TSV} is the TSV capacitance. Π -model is used to convert TSV RC into an equivalent RC model. In the above equation, buffers are not considered. If buffer insertion is taken into account, the delay becomes as follows:

$$\begin{aligned} D'_{PV} = & \frac{1}{2} \cdot \frac{R_{drv}}{S_{buf}} \cdot C_{TSV} \cdot N_{TSV} \\ & + \left(\frac{R_{drv}}{S_{buf}} + R_{TSV} \right) \cdot \left(\frac{1}{2} \cdot C_{TSV} + C_{drv} \right) \cdot N_{TSV} \\ & + (N_{TSV} - 1) \cdot D_{buf} \end{aligned} \quad (34)$$

Delay computation for *NPV* wires whose horizontal length is not negligible [32] is computed by combining Equation (31), (32), and Π -model of TSV RC with wire RC model.

Table 11. Comparison of the maximum delay. ‘B.I.’ means ‘Buffer Insertion’. BIS1 is the Buffer Insertion Scheme 1, and BIS2 is the Buffer Insertion Scheme 2 shown in Figure 24. ‘# B’ is the number of buffers. (Design : # gates = 40M, # dies = 4, and # signal TSVs = 8.3M).

TSV cap. (fF)	Delay of a 2D design [21] (longest WL = 14.6mm)			Delay of a 3D design (longest WL = 9.0mm)							
	w/o B.I.	BIS1	# B	Without TSV RC [33]			With TSV RC				
				w/o B.I.	BIS1	# B	w/o B.I.	BIS1	# B	BIS2	# B
5	61.3 ns	5.29 ns	9.8M	23.5 ns	3.24 ns	4.94M	23.5 ns	9.22 ns	11.8M	3.56 ns	20.1M
20							24.0 ns	9.23 ns		3.57 ns	
50							25.4 ns	9.24 ns		3.61 ns	
(column #)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
Delay ratio (w/o B.I.)	1.00			0.38			0.38 0.39 0.41				
Delay ratio (with B.I.)		1.00			0.61			1.74 1.74 1.75		0.67 0.67 0.68	
Buffer count ratio			1.00			0.50			1.20		2.05

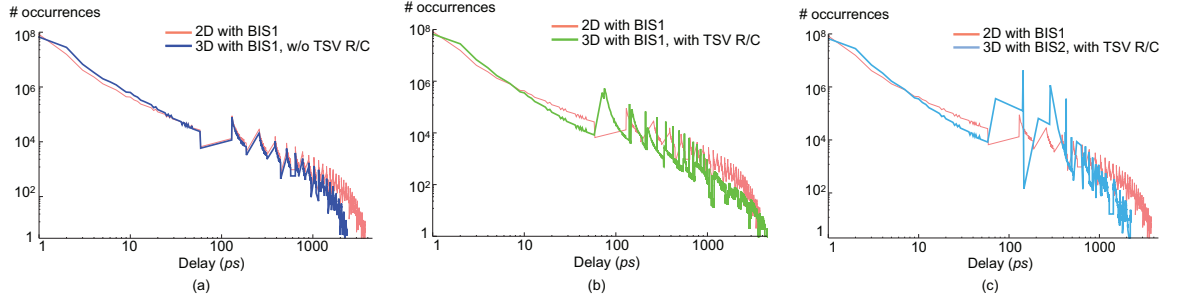


Figure 25. Delay distribution of 3D ICs (40M gates). (a) 3D with BIS1, w/o TSV RC, (b) 3D with BIS1, with TSV RC, and (c) 3D with BIS2, with TSV RC. TSV capacitance is 5fF.

2.6.5 Simulation Results

2.6.5.1 Maximum Delay and Buffer Count

The first experiment is on maximum delay and buffer counts. Table 11 compares the maximum delay and buffer counts of 2D and 3D ICs. In case of 2D ICs, the 2D wirelength prediction model in [21] is used for comparison. The 3D IC prediction is performed using the TSV-aware 3D wirelength prediction model in [33], where TSV RC parasitics are not considered.

First, columns (2), (5), and (8) in Table 11 compare the maximum delay without buffer insertion. The longest wire in 2D is about 14.6mm so that the delay in 2D is very high (61.3ns) without buffer insertion. On the other hand, the longest wire in 3D is about 9.0mm so that the delay without TSV RC in 3D is much smaller (23.5ns) than the 2D delay (61.3ns). Impact of TSV RC is shown in column (8) in Table 11. The maximum

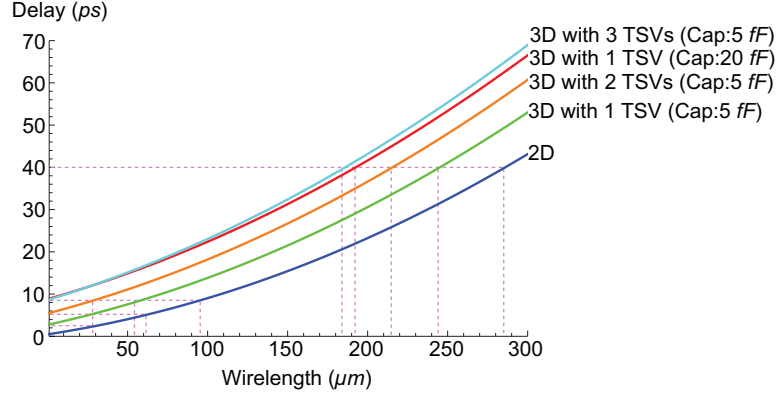


Figure 26. Comparison of delay in various cases. Wirelength in the x -axis does not include TSV height, and it is assumed that TSVs are evenly distributed along a 3D net. The driver size is $20\times$, and buffer insertion was not used.

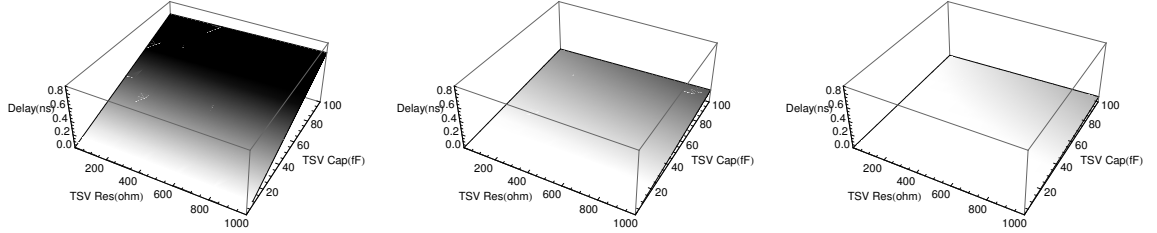


Figure 27. TSV RC vs Delay for each buffer size. (a) $1\times$ buffer, (b) $5\times$ buffer, (c) $20\times$ buffer

delay increases as TSV capacitance goes up. TSV resistance shows a similar trend, but the results are not shown in the table for brevity. However, the effect of TSV RC on the longest net is not significant because wire RC is much larger than that of TSVs in long nets.

Next, columns (3), (6), (9), and (11) in Table 11 compare the maximum delay with buffer insertion. The columns (4), (7), (10), and (12) compare the buffer usage for these cases. The maximum delay in 2D becomes $5.29ns$ after buffer insertion. On the other hand, the maximum delay in 3D becomes $3.24ns$ without TSV RC after buffer insertion. The difference ($2.05ns$) is quite significant because the longest wire ($9.0mm$) in 3D is much shorter than that ($14.6mm$) in 2D. The buffer count ($4.94M$) in 3D is also much smaller than the buffer count in 2D ($9.8M$).

If TSV RC is considered during delay computation, the maximum delay ($\sim 9.2ns$) becomes much bigger than the case without TSV RC ($3.24ns$). Moreover, the maximum delay is even bigger than the 2D case ($5.29ns$), and the buffer count ($11.8M$) increases

Table 12. Comparison of the maximum delay and buffer counts in different circuit sizes. TSV resistance is 100Ω and TSV capacitance is $20fF$. # dies = 4.

circuit area	# gates	longest wire		2D			3D w/o TSV RC			3D with TSV RC				
		2D	3D	w/o B.I.	BIS1	# B	w/o B.I.	BIS1	# B	w/o B.I.	BIS1	# B	BIS2	# B
400 mm ²	160M	29 mm	18 mm	243 ns	10.7 ns	64.2M	94.9 ns	6.60 ns	33M	95.6 ns	30.6 ns	62M	6.86 ns	97M
225 mm ²	90M	22 mm	14 mm	137 ns	7.98 ns	29.6M	53.1 ns	4.92 ns	15.3M	53.9 ns	18.5 ns	31M	5.18 ns	50M
100 mm ²	40M	15 mm	9 mm	61 ns	5.29 ns	9.78M	23.5 ns	3.24 ns	4.94M	24.0 ns	9.23 ns	12M	3.57 ns	20M
25 mm ²	10M	7.3 mm	4.4 mm	15.6 ns	2.61 ns	1.38M	5.84 ns	1.57 ns	0.66M	6.11 ns	3.12 ns	2.28M	2.00 ns	4.17M
1 mm ²	0.4M	1.5 mm	0.8 mm	0.70 ns	0.52 ns	7.09K	0.25 ns	0.27 ns	1.90K	0.31 ns	0.45 ns	51.7K	0.45 ns	103K
(column #)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)
Ratio of delay w/o B.I.	160M			1.00			0.39			0.39				
	90M			1.00			0.39			0.39				
	40M			1.00			0.39			0.39				
	10M			1.00			0.37			0.39				
	0.4M			1.00			0.36			0.44				
Ratio of delay with B.I.	160M				1.00			0.62			2.86		0.64	
	90M				1.00			0.62			2.32		0.65	
	40M				1.00			0.61			1.74		0.67	
	10M				1.00			0.60			1.20		0.77	
	0.4M				1.00			0.52			0.87		0.87	
Ratio of buffer count	160M					1.00			0.51			0.97		1.51
	90M					1.00			0.52			1.05		1.69
	40M					1.00			0.51			1.23		2.04
	10M					1.00			0.48			1.65		3.02
	0.4M					1.00			0.27			7.29		14.5

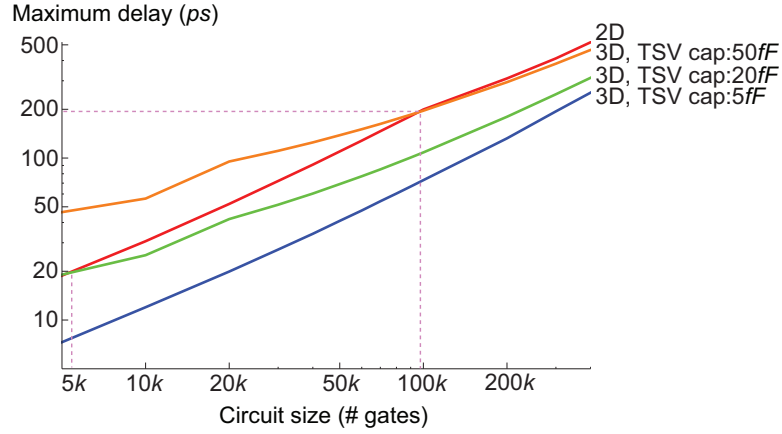


Figure 28. Circuit size vs maximum delay. # dies = 4.

significantly in BIS1 because a buffer is inserted right in front of a TSV. The increased maximum delay ($\sim 9.2ns$) is mainly due to C_{MAX} shown in Figure 24. Moreover, if a buffer is not inserted in front of a TSV, the maximum delay becomes much bigger than $9.2ns$.

In order to decrease the impact of TSV capacitance on the delay, a buffer is inserted as an intermediate sink at another end of a TSV in BIS2 as shown in Figure 24, and the maximum delay ($\sim 3.6ns$) finally becomes lower than the 2D case ($5.29ns$). However, the buffer count increases significantly ($20.1M$).

Figure 25 shows the delay distribution of the four cases (column 3, 6, 9, 11) in Table 11. As expected, there are more wires having small delay in 3D than 2D. In addition, the

Table 13. Total power (cell power + interconnect power + buffer internal power). unit:W. The ratios of 3D to 2D are shown in the parentheses.

Chip area (mm ²)	# gates	3D with TSV RC							
		2D	3D w/o TSV RC	TSV cap. = 5fF		TSV cap. = 20fF		TSV cap. = 50fF	
		BIS1	BIS1	BIS1	BIS2	BIS1	BIS2	BIS1	BIS2
25.0	10M	73.5	63.7(-13.3%)	68.7(-6.53%)	72.4(-1.50%)	78.1(+6.26%)	81.8(+11.3%)	92.3(+25.6%)	96.0(+30.6%)
12.5	5M	33.7	30.4(-9.79%)	32.7(-2.97%)	34.5(+2.37%)	37.2(+10.4%)	39.0(+15.7%)	43.9(+30.3%)	45.7(+35.6%)
2.50	1M	6.16	4.43(-28.1%)	5.98(-2.92%)	6.27(+1.79%)	6.72(+9.09%)	7.01(+13.8%)	7.82(+26.9%)	8.11(+31.7%)
1.25	500K	2.71	2.69(-0.74%)	2.88(+6.27%)	3.02(+11.4%)	3.21(+18.5%)	3.35(+23.6%)	3.71(+36.9%)	3.85(+42.1%)
0.25	100K	0.49	0.50(+2.04%)	0.53(+8.16%)	0.55(+12.2%)	0.58(+18.4%)	0.60(+22.4%)	0.65(+32.7%)	0.67(+36.7%)

Table 14. Additional silicon area (in mm²) required for buffer insertion)

Chip area (mm ²)	# gates	2D	3D w/o TSV RC	3D with TSV RC	
		BIS1	BIS1	BIS1	BIS2
250.3	100M	64.0	33.0	66.1	106.8
125.1	50M	25.0	12.7	28.9	48.6
25.0	10M	2.63	1.24	4.28	7.83
12.5	5M	1.4	0.43	1.90	3.57
2.50	1M	0.07	0.03	0.29	0.56
1.25	500K	0.02	0.01	0.13	0.25
0.25	100K	0.0004	0.0001	0.02	0.04

green graph in Figure 25(b), which shows ‘3D with BIS1, with TSV RC’, has the biggest delay because C_{MAX} in the case is the biggest capacitance which a buffer should drive. The discontinuities in Figure 25 are caused by buffer insertion.

2.6.5.2 Impact of TSV RC on Short and Medium Wires

The second experiment is on the impact of TSV RC on short and medium wires. Figure 26 shows delays in various cases for short/medium wires in 2D and 3D. First of all, there exists an intrinsic delay in a 3D net. This intrinsic delay cannot be decreased further because there exist TSVs in the net. Moreover, the intrinsic delay is strongly related to the number of TSVs used in a 3D net as well as TSV RC. For example, the intrinsic delay of ‘3D with 1 TSV’ is 3ps when the TSV capacitance is 5fF, and 9ps when the TSV capacitance is 20fF as shown in Figure 26. Similarly, the intrinsic delay of ‘3D with 3 TSVs’ is 9ps when the TSV capacitance is 5fF.¹

In addition, the minimum delay of ‘3D with 1 TSV (Cap:5fF)’ is 3ps, but this delay corresponds to a 30μm-long wire in 2D. Similarly, the delay of a short 3D wire having three

¹The intrinsic delay in Figure 26 is small because the driver size is 20×

TSVs (Cap: $5fF$) is $9ps$, but this delay corresponds to a $90\mu m$ -long wire in 2D. Therefore the following two cases have the same delay:

- Two gates are placed in 2D and their distance is $90\mu m$.
- One gate is placed in *die0*, another gate is placed in *die3* in 3D (so that there exist three TSVs), their horizontal distance is almost zero, and the TSV capacitance is $5fF$.

In order to benefit from 3D design for this net, the two gates need to be connected through less than three TSVs.

Figure 26 also shows that TSV RC needs to be considered in delay computation. According to the figure, using one TSV having bigger capacitance ($20fF$ for example) could be better than using three TSVs having smaller capacitance ($5fF$ for example) if TSVs are distributed evenly.

The impact of TSV RC on the delay for each buffer size is also presented in Figure 27. When the buffer size is small ($\sim 1\times$), the delay changes in a wide range ($10ps$ to $700ps$) as TSV capacitance varies from $1fF$ to $100fF$. On the other hand, TSV resistance does not have big impact on delay. For a medium-size buffer ($\sim 5\times$), TSV capacitance again has significant impact on delay, but the delay range ($2ps$ to $200ps$) is smaller than that of the $1\times$ buffer case. The impact of TSV resistance in this case becomes bigger if TSV capacitance is high. If the buffer size is big ($\sim 20\times$), the impact of TSV RC becomes small as shown in Figure 27(c).

2.6.5.3 Impact of TSV RC on Delay in Different Circuit Sizes

The third experiment is on the impact of TSV RC on delay in different circuit sizes. Table 12 compares the maximum delay and buffer counts for various circuit sizes. Columns (5), (8), and (11) in Table 12 compare the maximum delay without buffer insertion. In all the cases, the maximum delay of 3D ICs is much smaller than the maximum delay of 2D ICs even when TSV RC is considered. This is again mainly due to the fact that wire RC is

dominant in the longest wire. However, the difference between ‘2D w/o B.I.’ and ‘3D w/o B.I. with TSV RC’ becomes smaller as the circuit size goes down. 2D delay could eventually be smaller than 3D delay if the circuit size is very small. This means that 1) 2D ICs are superior to 3D ICs for small circuits, and 2) there exists a reversion point where 2D designs become better than 3D designs or vice versa. Since TSVs are used only in 3D ICs, the reversion point where 2D delay and 3D delay meet will increase as the TSV capacitance increases as shown in Figure 28. For example, if the TSV capacitance is $50fF$, the circuit should contain more than about $100k$ gates to benefit from 3D design.²

Columns (6), (9), (12), and (14) in Table 12 compare the maximum delay with buffer insertion, and columns (7), (10), (13), and (15) compare buffer counts. Similarly as shown in Table 11, buffer insertion schemes need to be considered carefully when TSV RC comes into the delay computation. Moreover, the buffer delay also needs to be taken into account. The delay of BIS1 and BIS2 (column 12,14) is worse than ‘w/o B.I.’ (column 11) for small circuits in our simulation because our buffer insertion scheme is not flexible.

2.6.5.4 Impact of TSV RC on Power

The fourth experiment is on the impact of TSV RC on power. Table 13 shows total chip power which consists of cell internal power, interconnect power, and buffer internal power. The power model presented in [34] is used to estimate interconnect power. Although the total wirelength becomes shorter in 3D ICs, the total power of 3D ICs could be greater than that of 2D ICs due to the non-negligible TSV capacitance and the number of buffers required to drive TSVs.

If TSV RC is ignored, power saving in 3D ICs is huge for medium-size or large circuits (−9% to −28%) because the total wirelength of 3D ICs is much smaller than the total wirelength of 2D ICs. However, if TSV RC is considered, power consumption of 3D ICs becomes bigger than 2D ICs unless TSV capacitance is small as shown in Table 13.

²The reversion point is dependent on TSV size, TSV RC, the number of dies, process technology, buffer insertion schemes, etc.

Therefore, power consumption of 3D ICs is expected to be greater than 2D ICs in general unless few TSVs are used in 3D ICs or TSV capacitance is small (e.g. less than $2fF$).

2.6.5.5 *Impact of Buffer Insertion on Silicon Area*

The fifth experiment is on the impact of buffer insertion on silicon area. Table 14 shows the additional silicon area required for buffer insertion. The additional area for buffer insertion in 2D ICs ranges from 2% for small circuits to 26% for big circuits compared to the original chip area. On the other hand, the additional area for 3D ICs ranges from 20% for small circuits to 40% for big circuits if BIS2 is used. This means that 3D IC is not suitable for too big circuits in terms of additional silicon area required for buffer insertion. Therefore, 3D ICs are suitable for medium or big circuits with respect to silicon area (Table 14), power consumption (Table 13), and the maximum delay (Figure 28) in the current assumptions and parameter settings.

2.7 Summary

This chapter presents TSV-aware analytical models predicting wirelength distribution of gate-level and block-level 3D ICs. A few parameters are newly introduced during the derivation of the models to explain characteristics of 3D ICs. The simulation results show that wirelength overhead caused by TSVs is not negligible, so the TSV count should be under control during TSV insertion in the design of 3D ICs. Early design exploration helping decision making for moving from 2D ICs to 3D ICs is also presented. With the TSV-aware wirelength distribution models, the impact of TSV parasitic RC on delay and power consumption of 3D ICs is also studied. The simulation results show that TSV capacitance is not negligible and it affects delay and power consumption of 3D ICs significantly. Since the impact of TSV RC on delay and power consumption of 3D ICs is not negligible, 3D designs could be worse than 2D designs unless buffers are inserted properly and the TSV count is controlled well. Therefore, proper buffer insertion algorithms for 3D ICs need to be developed considering non-negligible TSV RC. In addition, TSV-count-aware physical

design algorithms for 3D ICs also need to be developed in order to minimize side-effects of TSV RC.

CHAPTER 3

ANALYTICAL MODELING OF THROUGH-SILICON-VIA CAPACITIVE COUPLING

Driven by the need for performance improvement, a large number of universities and companies are actively researching three-dimensional integrated circuit (3D IC), which is expected to lead to shorter total wirelength, higher clock frequency, and lower power consumption than 2D IC [33, 2, 11]. In 3D IC, multiple dies are stacked, and vertical interconnections between dies are realized by through-silicon vias (TSVs). These TSVs play a central role in replacing long interconnects found in 2D ICs with short vertical interconnects. Shortened wires will result in lower wire delay, thereby improving performance. In addition, it is also possible with 3D heterogeneous integration to stack disparate technologies to provide a 3D structure with heterogeneous functions including logic, memory, MEMS, antennas, display, RF, analog/digital, sensors, and power conversion and storage. Therefore universities and companies have been actively developing TSV manufacturing and die-to-die bonding technologies [35, 36, 37, 38, 39, 40]. Moreover, various work on utilizing TSVs for physical design has also been proposed recently [41, 42].

The basic electrical characteristics of TSVs such as resistance, capacitance, and inductance have also been investigated in the literature to provide circuit designers with physical analyses of TSVs and ranges of their values [43, 44, 45, 46, 30]. One of the results to notice is that TSV coupling capacitance is very big (tens of femto-farads) [44] so that it has huge impact on timing and interconnect power [32, 47]. Therefore, computer-aided design (CAD) tools are required to compute TSV coupling capacitance quickly but accurately during placement, routing, and optimization of timing and power in 3D ICs.

TSV-to-TSV (or TSV-to-wire) coupling capacitance is affected by TSV-to-TSV (or TSV-to-wire) distance, TSV and wire dimensions, the number of surrounding TSVs and wires, and their spatial distribution. It is therefore almost impossible to use look-up tables

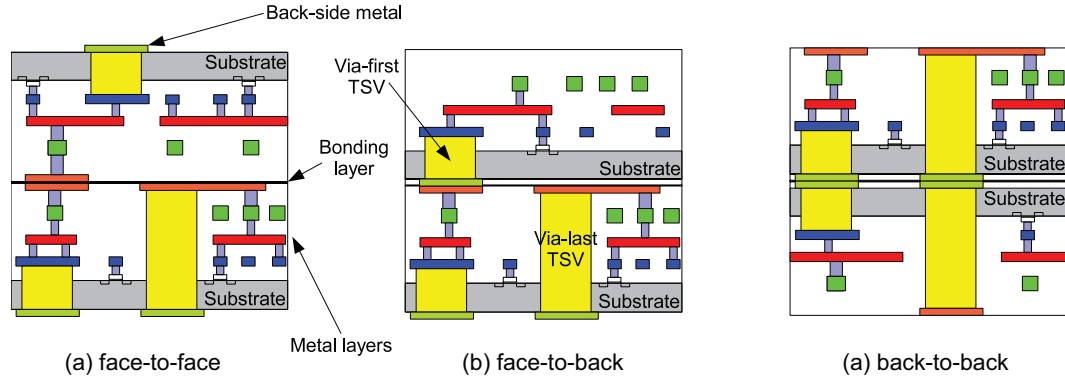


Figure 29. Three types of die bonding (face-to-face, face-to-back, and back-to-back) and two types of TSVs (via-first and via-last).

to compute TSV capacitance quickly because too many variables exist. In addition, it is also almost impossible to use field solvers for TSV capacitance computation during placement, routing, or optimization of timing and power because field solvers require non-negligible amount of computation time.

3.1 Preliminaries

3.1.1 TSV Formation and Die Bonding

Figure 29 shows three types of die bonding and two types of TSVs. Under the via-first technology, devices and TSVs are fabricated first, metal layers are deposited, and then dies are bonded. Therefore, TSVs in via-first technology are surrounded by other TSVs laterally and by wires vertically. In via-last technology, on the other hand, devices and metal layers are fabricated first, TSVs are fabricated through all the layers from the substrate to the topmost metal layer, and then dies are bonded. Therefore, TSVs in via-last technology are surrounded by other TSVs laterally and by wires laterally and vertically.

3.1.2 TSV Coupling Capacitance

TSV coupling capacitance consists mainly of two components as follows:

- Capacitive coupling (C_{TW} in Figure 30) between a TSV and wires surrounding the TSV. These wires exist on top or bottom of TSVs in via-first case as shown in the figure. In case of via-last TSVs, there exists capacitive coupling between a TSV and

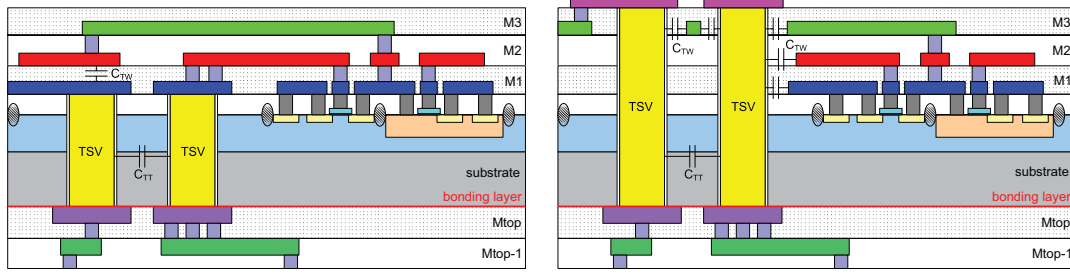


Figure 30. Left: Capacitive coupling in via-first TSV technology. Right: Capacitive coupling in via-last TSV technology.

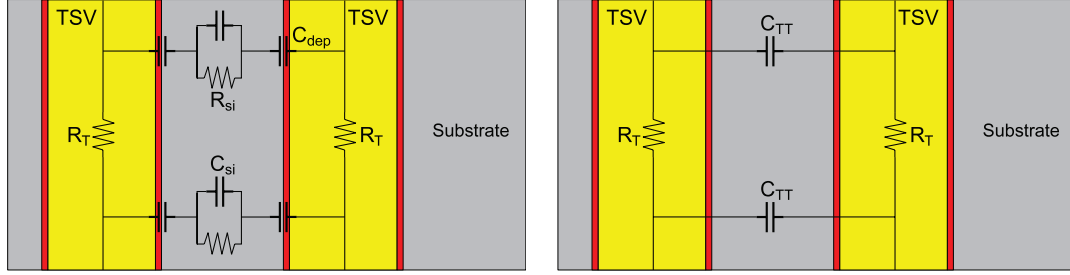


Figure 31. Left: TSV RC model. Right: Simplified TSV RC model.

neighboring wires in metal layers.

- Capacitive coupling (C_{TT} in Figure 30) between two TSVs.

To analyze physical phenomena between two TSVs in the substrate, previous models presented in the literature are reviewed. Figure 31(a) shows a TSV RC model presented in [46, 48] ¹. In the model, two TSVs are connected by a series connection of C_{dep} , a parallel connection of C_{si} and R_{si} , and C_{dep} . The impedance of the parallel connection of C_{si} and R_{si} is as follows:

$$Z_{si} = \frac{R_{si}}{1 + j\omega R_{si} C_{si}} \quad (35)$$

where C_{si} and R_{si} are capacitance and resistance of the silicon substrate respectively. If the substrate is pure silicon substrate or high-resistivity substrate (HRS) so that R_{si} is high, Z_{si} in Equation (35) is determined primarily by C_{si} . In this case, this model can be simplified by removing R_{si} . The simplified model is shown in Figure 31(b) which is the interest of this project ².

¹The model shown here is a simplified model obtained by ignoring TSV inductance.

²If the substrate resistivity is low, substrate resistance should not be ignored in Equation (35).

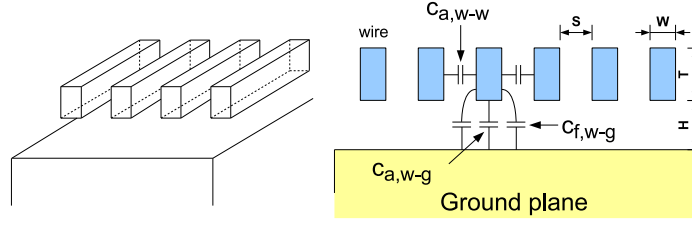


Figure 32. Capacitance of multiple wires on ground plane.

In this simplified model, the liner capacitance between a TSV and the silicon substrate is also ignored. The reason is because it is assumed that the liner is very thin so that the liner capacitance is very high compared to the TSV-to-TSV coupling capacitance, and the focus of this project is on high frequency ranges. If more accurate models are required, capacitance formulas presented in [30] can be used for the liner capacitance computation.

3.1.3 Basic Formulas for Capacitance Computation

3.1.3.1 Multiple Wires on Ground Plane

In 3D IC layouts, multiple wires go over a TSV which can be considered as a ground plane. Therefore, capacitance formulas for multiple wires laid on a ground plane are reviewed.

Figure 32 shows the side view of wires and a ground plane, and [49] shows capacitance formulas for multiple wires on a ground plane as follows:

$$c_{a,w-g} = \epsilon_{di} \cdot 1.15 \left(\frac{W}{H} \right) \quad (36)$$

$$c_{f,w-g} = \epsilon_{di} \cdot 2.80 \left(\frac{T}{H} \right)^{0.222} \quad (37)$$

$$c_{w-g} = c_{a,w-g} + 2 \cdot c_{f,w-g} \quad (38)$$

$$C_{w-g} = L_{wire} \cdot c_{wire} \quad (39)$$

where W is the wire width, T is the wire thickness, H is the spacing between a wire and the ground plane, $c_{a,w-g}$ is the area capacitance³ per unit length between the bottom surface of the wire and the top surface of the ground plane, $c_{f,w-g}$ is the fringe capacitance per unit length between a sidewall of the wire and the top surface of the ground plane, and ϵ_{di} is the dielectric constant of the dielectric material. c_{w-g} , the coupling capacitance per unit length

³Area capacitance is the capacitance between two parallel plates.

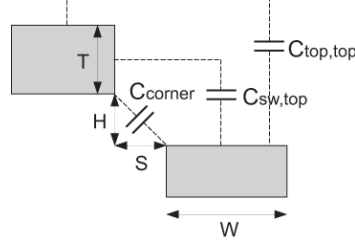


Figure 33. Various fringe capacitances.

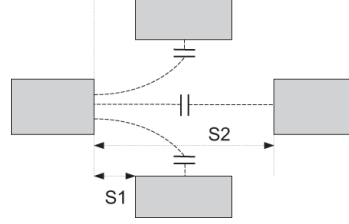


Figure 34. Fringe capacitances when surrounding wires exist.

between a wire and the ground plane, is the sum of one area capacitance and two fringe capacitances as shown in Figure 32. C_{w-g} is the final total coupling capacitance between a wire and the ground plane when multiple wires exist.

3.1.3.2 Fringe Capacitance

Formulas of fringe capacitances between two wires are presented in [50], and the geometry and formulas are repeated in Figure 33 and Equation (40)-(44).

$$c_{sw,top} = \frac{\epsilon_{di}}{\pi/2} \cdot \ln\left[\frac{H + \eta T + \sqrt{S^2 + (\eta T)^2 + 2H\eta T}}{S + H}\right] \quad (40)$$

$$c_{top,top} = \frac{\epsilon_{di} W \alpha (\ln[1 + \frac{2W}{S}] + e^{(-\frac{S+T}{3S})})}{W \pi \alpha + (H + T)(\ln[1 + \frac{2W}{S}] + e^{(-\frac{S+T}{3S})})} \quad (41)$$

$$c_{corner} = \frac{\epsilon_{di}}{\pi} \sqrt{\frac{HS}{H^2 + S^2}} \quad (42)$$

$$\eta = \exp[(W + S - \sqrt{S^2 + T^2 + 2HT})/(\tau W)]$$

$$\alpha = \exp[-(H + T)/(S + W)] \quad (43)$$

$$\eta_0 = \exp\left[-\frac{\sqrt{S_1^2 + (H + \frac{1}{2}T)^2} + S_1}{2S_2} + \frac{1}{5}\right] \quad (44)$$

where W is the metal width, T is the metal thickness, H is the vertical spacing, and S is the horizontal spacing. $c_{sw,top}$ is the capacitance per unit length between the sidewall of the

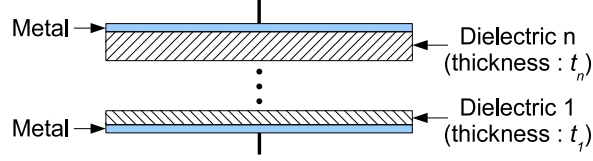


Figure 35. Multiple dielectric materials in a parallel plate capacitor.

upper wire and the top surface of the lower wire. $c_{\text{top,top}}$ is the capacitance per unit length between the top surfaces of the upper and lower wire. c_{corner} is the capacitance per unit length between the two corners. If there are surrounding wires as shown in Figure 34, it is necessary to multiply $c_{\text{sw,top}}$ by η_0 shown in Equation (44) to account for new distribution of electric field [50].

3.1.3.3 Multiple Dielectric Materials

When multiple dielectric materials exist between two parallel plates as shown in Figure 35, its capacitance is computed by the following equation:

$$C = \epsilon_0 \cdot \epsilon_{\text{new}} \cdot \frac{S}{\sum_{i=1}^n t_i} \quad (45)$$

$$\epsilon_{\text{new}} = \left(\sum_{i=1}^n t_i \right) \cdot \left(\sum_{j=1}^n \frac{t_j}{\epsilon_{r,j}} \right)^{-1} \quad (46)$$

where ϵ_0 is the vacuum permittivity, ϵ_{new} is the relative permittivity of the parallel plate capacitor, S is the area of the parallel plate, n is the number of dielectric layers, t_i is the thickness of i -th dielectric layer, and $\epsilon_{r,j}$ is the dielectric constant of j -th dielectric layer [51].

3.1.3.4 Capacitance between Two Surfaces

$c_{\text{sw,top}}$ in Equation (40) is valid when two wires are in the geometric relation shown in Figure 33. If two surfaces are not in this geometric relation, however, $c_{\text{sw,top}}$ cannot be applied directly to compute the coupling capacitance of the two surfaces. Figure 36 shows an example where the geometric relation between the two surfaces F1 and F2 is different from the geometric relation in Figure 33.

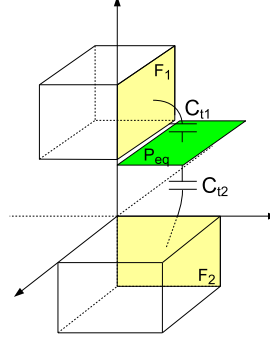


Figure 36. Capacitance between two surfaces.

In this case, a simple approximation technique is used as follows. First, a flat equipotential plane is found between the two metal surfaces. Then, the coupling capacitance between a metal surface and the equipotential plane (C_{t1} and C_{t2} in the figure) is computed. Finally, the coupling capacitance between the two metal surfaces is computed by the series connection of the two coupling capacitances. In Figure 36, for example, the coupling capacitance between two metal surfaces F_1 and F_2 is computed by assuming the equipotential plane P_{eq} and computing C_{t1} and C_{t2} using $c_{sw,top}$. The final coupling capacitance between F_1 and F_2 is the capacitance of the series connection of C_{t1} and C_{t2} .

To validate the approximation, capacitance computation by this technique is applied to several randomly-generated geometries and its results are compared against Raphael [31] simulation. The error is around 10% but this is tolerable because absolute values of this kind of fringe capacitance are much smaller than TSV-to-TSV coupling capacitance or TSV-to-wire area capacitance.

3.2 Analytical Modeling of TSV Capacitance

3.2.1 TSVs with Top and Bottom Neighbors

One of major challenges in the computation of TSV-related capacitances is in identifying different capacitive components. Therefore, all capacitive components in a regular TSV structure are identified in this section. A TSV in the regular TSV structure is surrounded by eight other TSVs and top (and bottom) wires as shown in Figure 37 (a). Table 15 shows variables and constants used in the formulation of capacitances.

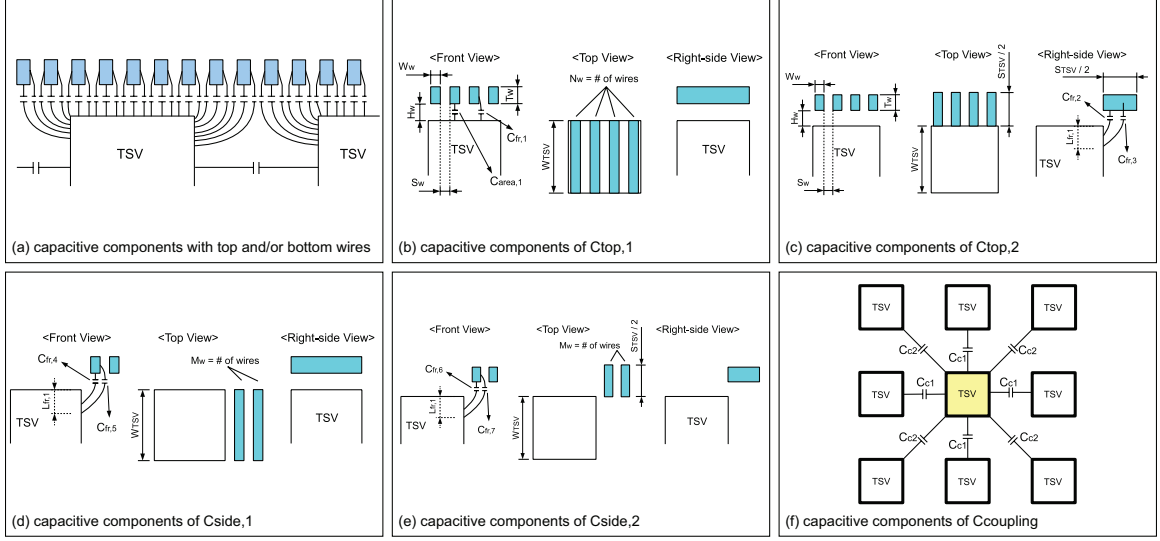


Figure 37. Capacitive components of TSVs with top and bottom neighboring wires.

3.2.1.1 Modeling $C_{top,1}$

$C_{top,1}$ is the capacitance between the top surface of a TSV and the wires on top of the TSV as shown in Figure 37 (b). Table 16 shows the variable settings for $C_{area,1}$ and $C_{fr,1}$. $C_{top,1}$ is computed as follows:

$$\begin{aligned}
 C_{area,1} &= c_{area,1} \cdot W_{TSV} \\
 C_{fr,1} &= c_{fr,1} \cdot W_{TSV} \\
 C_{top,1} &= N_w \cdot (C_{area,1} + 2C_{fr,1})
 \end{aligned} \tag{47}$$

where $C_{area,1}$ is the coupling capacitance between the bottom surface of the wires and the top surface of the TSV, $C_{fr,1}$ is the coupling capacitance between the sidewalls of wires and the top surface of the TSV. $c_{area,1}$ is computed by plugging W_w , S_w , and H_w into W , S , and H respectively in Equation (36), and plugging $\frac{S_w}{2}$, 0 , T_w , H_w , 0 , and S_w into W , S , T , H , S_1 , and S_2 respectively in Equation (40) and Equation (44). Table 16 shows these substitution settings.

3.2.1.2 Modeling $C_{top,2}$

$C_{top,2}$ is the capacitance between a sidewall of the TSV and the outside wire pieces which are actually connected to wires on top of the TSV as shown in Figure 37 (c). $L_{fr,1}$ is

Table 15. Variables and constants used in capacitance extraction

W_{TSV}	TSV width (assuming TSVs are square-shaped)
H_{TSV}	TSV height (length in z-direction)
S_{TSV}	spacing between two TSVs
W_w	metal wire width
S_w	spacing between two parallel metal wires
T_w	metal wire thickness
H_w	spacing in z-direction between two adjacent metal layers
N_w	the number of wires on top (or bottom) of a TSV ($= \frac{W_{\text{TSV}}}{W_w + S_w}$)
M_w	a half the number of wires between two TSVs ($= \frac{S_{\text{TSV}}}{2(W_w + S_w)}$)
$L_{\text{fr},1}$	effective length affecting fringe capacitance of a TSV ($= 0.4 \cdot \frac{S_{\text{TSV}}}{2}$) (see Figure 37 (c), (d), (e))
$L_{\text{fr},2}$	effective length affecting fringe capacitance of a TSV ($= 0.4 \cdot \frac{S_{\text{TSV}} - 2 \cdot S_{\text{min}}}{2}$) (see Figure 38 (d), (e))
S_{min}	minimum spacing between a metal wire and a TSV (see Figure 38 (d), (e))
M'_w	$= \frac{S_{\text{TSV}} - 2 \cdot S_{\text{min}}}{2(W_w + S_w)}$ (see Figure 38 (e))
H_{INT}	height of interconnect layers between TSVs (see Figure 38 (a))
W_{mis}	$= W_{\text{TSV}} \cdot \text{misalignment ratio.}$ (see Figure 38 (f))

determined empirically, and Table 16 shows the variable settings for $C_{\text{fr},2}$ and $C_{\text{fr},3}$. $C_{\text{top},2}$ is computed as follows:

$$\begin{aligned}
 C_{\text{fr},2} &= c_{\text{fr},2} \cdot W_w \\
 C_{s1} &= c_{s1} \cdot \frac{S_{\text{TSV}}}{2}, \quad C_{s2} = c_{s2} \cdot \frac{S_w}{2}, \quad C_{\text{fr},3} = C_{s1} // C_{s2} \\
 C_{\text{top},2} &= N_w \cdot [C_{\text{fr},2} + 2 \cdot C_{\text{fr},3}]
 \end{aligned} \tag{48}$$

where $C_{\text{fr},2}$ is the coupling capacitance between the bottom side of a wire and a sidewall of the TSV, and $C_{\text{fr},3}$ is the coupling capacitance between sidewalls of wires and a sidewall of the TSV.

Table 16. Variable settings. C.F. means ‘capacitance function’. Series means the components are connected in series (e.g., $c_{fr,3}$ is computed by the series connection of c_{s1} and c_{s2} .)

		Series	C.F.	W	S	T	H	S_1	S_2
$C_{top,1}$	$c_{area,1}$		$c_{a,w-g}$	W_w	-	-	H_w	-	-
	$c_{fr,1}$		$c_{f,w-g}$	-	-	T_w	H_w	-	-
$C_{top,2}$	$c_{fr,2}$		$c_{sw,top}$	$L_{fr,1}$	H_w	$\frac{S_{TSV}}{2}$	0	-	-
	$c_{fr,3}$	c_{s1}	$c_{sw,top}$	$\frac{S_w}{2}$	0	T_w	$\frac{H_w}{2}$	0	S_w
		c_{s2}	$c_{sw,top}$	$L_{fr,1}$	$\frac{H_w}{2}$	$\frac{S_{TSV}}{2}$	0	-	-
$C_{side,1}$	$c_{fr,4}(m)$		$c_{sw,top}$	$\frac{L_{fr,1}}{2 \cdot M_w}$	$H_w + (2m-1) \frac{L_{fr,1}}{2 \cdot M_w}$	W_w	$m \cdot S_w + (m-1)W_w$	-	-
	$c_{fr,5}(m)$		$c_{top,top}$	$\frac{L_{fr,1}}{4 \cdot M_w}$	$H_w + m \frac{L_{fr,1}}{M_w}$	W_w	$m \cdot S_w + (m-1)W_w$	-	-
$C_{side,2}$	$c_{fr,6}(m)$	c_{s3}	$c_{sw,top}$	$\frac{S_{TSV}}{2}$	H_w	$\frac{S_{TSV}}{2}$	0	-	-
		$c_{s4}(m)$	$c_{sw,top}$	$\frac{L_{fr,1}}{2 \cdot M_w}$	$S_w + (2m-1) \frac{L_{fr,1}}{2 \cdot M_w}$	W_w	$m \cdot S_w + (m-1)W_w$	-	-
	$c_{fr,7}(m)$	c_{s5}	$c_{sw,top}$	$\frac{S_w}{2}$	0	T_w	H_w	0	S_w
		c_{s6}	$c_{sw,top}$	$\frac{S_{TSV}}{2}$	H_w	$\frac{S_{TSV}}{2}$	0	-	-
		c_{s7}	$c_{sw,top}$	$\frac{L_{fr,1}}{2 \cdot M_w}$	$S_w + (2m-1) \frac{L_{fr,1}}{2 \cdot M_w}$	S_w	$m \cdot S_w + (m-1)W_w$	-	-
$C_{side,3}$	$c_{area,2}$		$c_{a,w-g}$	W_w	-	-	S_{min}	-	-
	$c_{sw,1}$		$c_{sw,top}$	$\frac{S_w}{2}$	0	$\frac{S_w}{2}$	S_{min}	0	S_w
	$c_{sw,2}$		$c_{sw,top}$	$\frac{H_w}{2}$	0	W_w	S_{min}	0	H_w
$C_{side,4}$	$c_{area,3}$		$c_{a,w-g}$	T_w	-	-	S_{min}	-	-
	$c_{sw,3}$		$c_{sw,top}$	$\frac{H_w}{2}$	0	W_w	S_{min}	0	H_w
$C_{side,5}$	$c_{sw,4}$		$c_{sw,top}$	$L_{fr,2}$	0	$\frac{S_{TSV}-2S_{min}}{2}$	S_{min}	-	-
$C_{side,6}$	$c_{sw,5}(m)$		$c_{sw,top}$	$\frac{L_{fr,2}}{2M'_w}$	$\frac{(2m-1)L_{fr,2}}{2M'_w}$	W_w	$S_{min} + (m-1)W_w + (m-1)S_w$	-	-
	$c_{sw,6}(m)$		$c_{top,top}$	$\frac{L_{fr,2}}{4M'_w}$	$m \frac{L_{fr,2}}{M'_w}$	W_w	$S_{min} + m \cdot W_w + (m-1)S_w$	-	-
C_{m2}			$c_{sw,top}$	W_{mis}	$S_{TSV}-W_{mis}$	W_{mis}	0	-	-

3.2.1.3 Modeling $C_{side,1}$

$C_{side,1}$ is the capacitance between a sidewall of the TSV and side wires as shown in Figure 37 (d). Table 16 shows the variable settings for $C_{side,1}$. $C_{side,1}$ is computed as follows:

$$\begin{aligned}
 C_{fr,4}(m) &= c_{fr,4}(m) \cdot W_{TSV} \\
 C_{fr,5}(m) &= c_{fr,5}(m) \cdot W_{TSV} \\
 C_{side,1} &= \sum_{m=1}^{M_w} (C_{fr,4}(m) + 2 \cdot C_{fr,5}(m))
 \end{aligned} \tag{49}$$

where $C_{fr,4}(m)$ is the coupling capacitance between the bottom side of the m -th wire and the facing wall of the TSV, and $C_{fr,5}(m)$ is the coupling capacitance between the sidewalls of the m -th wire and the facing wall of the TSV.

3.2.1.4 Modeling $C_{side,2}$

$C_{side,2}$ is the capacitance between a sidewall of the TSV and side wires in non-overlapped regions as shown in Figure 37 (e). Table 16 shows the variable settings for $C_{side,2}$. $C_{side,2}$ is

computed as follows:

$$\begin{aligned}
C_{s3} &= c_{s3} \cdot W_w, \quad C_{s4}(m) = c_{s4}(m) \cdot \frac{S_{\text{TSV}}}{2} \\
C_{fr,6}(m) &= C_{s3} / C_{s4}(m) \\
C_{s5} &= c_{s5} \cdot \frac{S_{\text{TSV}}}{2}, \quad C_{s6} = c_{s6} \cdot S_w \\
C_{s7}(m) &= c_{s7}(m) \cdot \frac{S_{\text{TSV}}}{2} \\
C_{fr,7}(m) &= C_{s5} / C_{s6} / C_{s7}(m) \\
C_{\text{side},2} &= \sum_{m=1}^{M_w} [C_{fr,6}(m) + 2 \cdot C_{fr,7}(m)] \tag{50}
\end{aligned}$$

where $C_{fr,6}(m)$ is the coupling capacitance between the bottom side of the m -th wire and the facing sidewall of the TSV, and $C_{fr,7}(m)$ is the coupling capacitance between sidewalls of the m -th wire and the facing sidewall of the TSV.

3.2.2 Modeling of TSV-to-TSV Coupling Capacitance

Capacitive coupling exists between two adjacent TSVs. This coupling capacitance C_{TT} between two TSVs consists of two components. The first component is the coupling capacitance (C_{c1} in Figure 37 (f)) between the sidewalls of the TSVs, and the second component is the coupling capacitance (C_{c2} in Figure 37 (f)) between the corners of the TSVs. C_{c1} is computed as follows:

$$C_{c1} = \epsilon_{\text{di}} \frac{(H_{\text{TSV}} - 2 \cdot L_{fr,1}) \cdot W_{\text{TSV}}}{S_{\text{TSV}}} \tag{51}$$

c_{corner} in Equation (42) which will be used for the computation of C_{c2} is dependent on S/H . If H and S are constants, c_{corner} also becomes a constant. In our case, however, the width, height, and spacing of TSVs vary in a wide range. Therefore, a proportional constant, K_{corner} , is empirically found, and C_{c2} is computed as follows:

$$C_{c2} = \frac{\epsilon_{\text{di}}}{\pi \sqrt{2}} \cdot H_{\text{TSV}} \cdot K_{\text{corner}} \tag{52}$$

$$\begin{aligned}
K_{\text{corner}} &= \frac{1}{2} \cdot \frac{H_{\text{TSV}}}{S_{\text{TSV}}} \quad (\text{if } \frac{H_{\text{TSV}}}{S_{\text{TSV}}} \leq 4.0) \\
&= 2.0 \quad (\text{if } \frac{H_{\text{TSV}}}{S_{\text{TSV}}} \geq 4.0) \tag{53}
\end{aligned}$$

Lastly, C_{TT} is computed by the following equation:

$$C_{TT} = 4(C_{c1} + C_{c2}) \quad (54)$$

3.2.2.1 Impact of TSV Liner

It is required to consider multiple dielectric materials when TSV-to-wire fringe capacitance or TSV-to-TSV coupling capacitance is computed because multiple dielectric materials exist between two conductors. In this case, the capacitance formula shown in Equation (46) is used to take multiple dielectrics into account. In our simulation, the impact of TSV liner is neglected because it is assumed that TSV liner is very thin (approximately $0.1\mu\text{m}$) compared to TSV-to-wire or TSV-to-TSV distance, thus ε_{new} in Equation (46) is dominated mainly by ILD and substrate. If TSV liner thickness is not negligible, however, Equation (46) needs to be applied so that multiple dielectric materials are considered.

3.2.2.2 Metal Wires Connected to TSVs

If a metal wire on top of a TSV is connected to the TSV in Figure 37 (a), the coupling capacitance between the wire and the TSV should be subtracted from the TSV capacitance. In this case, however, wire-to-wire coupling capacitances ($c_{a,w-w}$) shown in Figure 32 needs to be added to the TSV capacitance. The wire-to-wire coupling capacitance is computed by the following formula [49]:

$$c_{a,w-w} = \varepsilon_{\text{di}} \cdot \left(0.03\left(\frac{W}{H}\right) + 0.83\left(\frac{T}{H}\right) - 0.07\left(\frac{T}{H}\right)^{0.222}\left(\frac{S}{H}\right)^{-1.34} \right) \quad (55)$$

where W is the wire width, T is the wire thickness, H is the spacing between a wire and the ground plane, and S is the spacing between two adjacent wires.

3.2.3 TSVs with Top, Bottom, and Side Neighbors

Figure 38 (a) shows capacitance components when a TSV is surrounded by neighboring wires vertically and laterally.

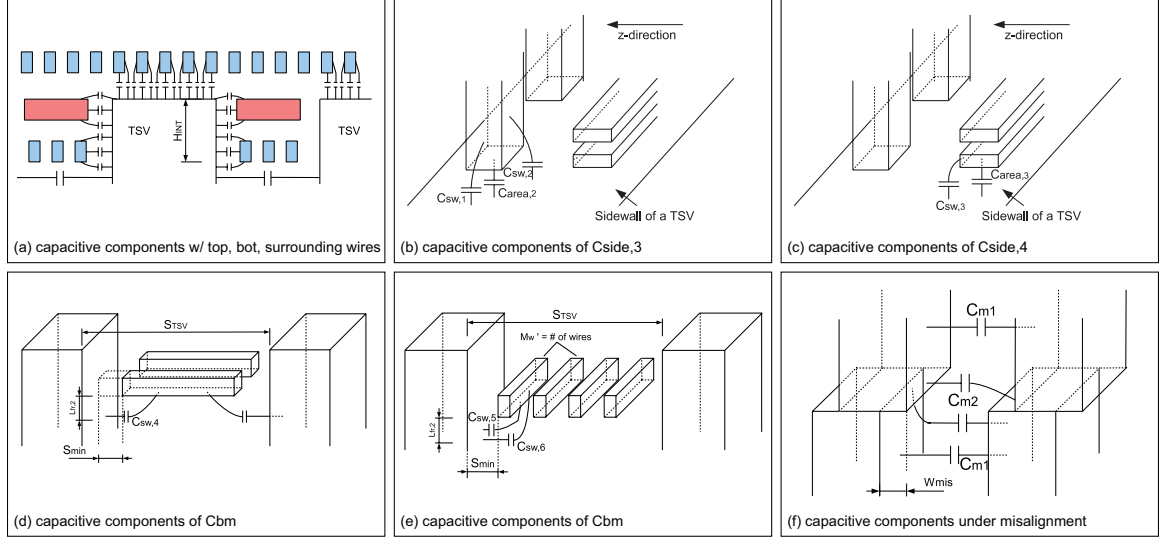


Figure 38. Capacitive components of TSVs with top, bottom, and side neighboring wires.

3.2.3.1 Modeling $C_{\text{side},3}$

$C_{\text{side},3}$ consists of three components $C_{\text{area},2}$, $C_{\text{sw},1}$, and $C_{\text{sw},2}$ as shown in Figure 38 (b). Table 16 shows the variable settings for these three components. $C_{\text{side},3}$ is computed as follows:

$$\begin{aligned}
 C_{\text{area},2} &= c_{\text{area},2} \cdot T_w \\
 C_{\text{sw},1} &= c_{\text{sw},1} \cdot T_w, \quad C_{\text{sw},2} = c_{\text{sw},2} \cdot W_w \\
 C_{\text{side},3} &= N_w \cdot (C_{\text{area},2} + 2 \cdot C_{\text{sw},1} + 2 \cdot C_{\text{sw},2})
 \end{aligned} \tag{56}$$

where $C_{\text{area},2}$ is the coupling capacitance between facing sidewalls of a wire and the TSV, and $C_{\text{sw},1}$ and $C_{\text{sw},2}$ are the coupling capacitances between a sidewall of a wire and the facing sidewall of the TSV.

3.2.3.2 Modeling $C_{\text{side},4}$

$C_{\text{side},4}$ consists of two components $C_{\text{area},3}$ and $C_{\text{sw},3}$ as shown in Figure 38 (c). Table 16 shows the variable setting for these two components. $C_{\text{side},4}$ is computed as follows:

$$\begin{aligned} C_{\text{area},3} &= c_{\text{area},3} \cdot W_{\text{TSV}} \\ C_{\text{sw},3} &= c_{\text{sw},3} \cdot W_{\text{TSV}} \\ C_{\text{side},4} &= C_{\text{area},3} + 2 \cdot C_{\text{sw},3} \end{aligned} \quad (57)$$

where $C_{\text{area},3}$ is the coupling capacitance between a sidewall of a wire and the facing sidewall of the TSV, and $C_{\text{sw},3}$ is the coupling capacitance between the top surface of a wire and the facing sidewall of the TSV.

3.2.3.3 Modeling C_{bm}

M1 layer has no additional metal layers below it, so the coupling capacitance C_{bm} between an M1 wire and a sidewall of a TSV is computed as follows:

$$\begin{aligned} C_{\text{sw},4} &= c_{\text{sw},4} \cdot W_{\text{w}} \\ C_{\text{side},5} &= N_{\text{w}} \cdot C_{\text{sw},4} \end{aligned} \quad (58)$$

$$\begin{aligned} C_{\text{sw},5}(m) &= c_{\text{sw},5}(m) \cdot W_{\text{TSV}} \\ C_{\text{sw},6}(m) &= c_{\text{sw},6}(m) \cdot W_{\text{TSV}} \\ C_{\text{side},6} &= \sum_{m=1}^{M'_{\text{w}}} (C_{\text{side},5} + 2 \cdot C_{\text{side},6}) \end{aligned} \quad (59)$$

$$C_{\text{bm}} = 2 \cdot (C_{\text{side},5} + C_{\text{side},6}) \quad (60)$$

where $C_{\text{sw},4}$ and $C_{\text{sw},5}(m)$ are the coupling capacitances between the bottom side of wires and the facing sidewall of the TSV, and $C_{\text{sw},6}(m)$ is the coupling capacitance between a sidewall of the m -th wire and the facing sidewall of the TSV as shown in Figure 38 (d) and (e). $L_{\text{fr},2}$ is determined empirically.

3.2.3.4 Modeling C_{TT}

Lastly, C_{TT} is computed as follows:

$$\begin{aligned} C_{c3} &= \epsilon_{di} \frac{(H_{TSV} - 2 \cdot H_{INT} - 2 \cdot L_{fr,2})W_{TSV}}{S_{TSV}} \\ C_{c4} &= \frac{\epsilon_{di}}{\pi \sqrt{2}} \cdot H_{TSV} \cdot K_{corner} \\ C_{TT} &= 4(C_{c3} + C_{c4}) \end{aligned} \quad (61)$$

where C_{c3} is the coupling capacitance between two TSVs placed in parallel, and C_{c4} is the coupling capacitance between two TSVs placed diagonally.

3.2.4 Modeling of Misalignment

Misalignment between TSVs occurs due to imperfectness of die aligning [52]. Therefore, TSV capacitance modeling under misalignment is shown in this section. Figure 38 (f) shows a model for misalignment. In this model, it is assumed that the capacitance near the bonding layer is not affected by surrounding wires of TSVs for simplification.

In Figure 38 (f), C_{m1} is computed by the area capacitance equation, and C_{m2} is computed by Table 16 and the following equation:

$$C_{m2} = c_{m2} \cdot W_{TSV} \quad (62)$$

where c_{m2} is the coupling capacitance between the top surface of a TSV and the facing sidewall of its neighboring TSV.

3.3 TSV Capacitance Extraction and Simulation

Synopsys Raphael simulation is run on a SUN UltraSPARC-II 400MHz machine with 4GB main memory. Wire width is $0.2\mu m$, wire thickness is $0.36\mu m$, wire-to-wire spacing is $0.2\mu m$, and wire-to-TSV spacing is $0.3\mu m$. Liner thickness is $0.1\mu m$.

3.3.1 TSVs with Top and Bottom Neighbors

The first comparison is on a structure composed of TSVs with top and bottom neighboring wires. The Raphael simulation structure for this comparison consists of nine TSVs forming

Table 17. Comparison of capacitances for TSVs with wires above and below the TSVs under perfect TSV-to-TSV alignment. The computation time of our model is negligible for all the cases. W is the TSV width, S is the TSV-to-TSV spacing, H is the TSV height, and R is the runtime of Raphael in minutes.

TSV dimension (μm)			TSV capacitance (fF)			Breakdown of capacitive components						R
W	S	H	Raphael	Our model	Error	Raphael			Our model			
						C_{TT}	C_{top}	C_{side}	C_{TT}	C_{top}	C_{side}	
5	5	5	8.868	9.389	5.88%	14.86%	53.3%	31.84%	15.72%	56.21%	28.07%	3
		20	18.336	19.102	4.18%	57.04%	26.12%	16.84%	58.57%	27.63%	13.8%	3
		50	37.033	37.129	0.26%	78.69%	12.94%	8.37%	78.69%	14.21%	7.10%	4
		100	68.227	67.174	-1.54%	88.36%	7.04%	4.60%	88.22%	7.86%	3.92%	4
	10	20	15.706	15.939	1.48%	29.17%	32.01%	38.82%	32.5%	39.07%	28.43%	6
		50	27.984	29.615	5.83%	60.15%	17.97%	21.88%	63.67%	21.02%	15.31%	6
100		48.437	49.301	1.78%	76.97%	10.39%	12.64%	78.18%	12.63%	9.19%	7	
10	10	10	26.079	27.210	4.34%	7.45%	62.55%	30.01%	10.85%	65.32%	23.83%	15
		20	32.645	32.752	0.33%	22.91%	50.59%	26.50%	25.94%	54.26%	19.80%	16
		50	51.392	52.644	2.44%	50.84%	32.19%	16.97%	53.92%	33.76%	12.32%	16
		100	82.570	82.689	0.14%	69.40%	20.03%	10.57%	70.66%	21.49%	7.85%	19
20	20	20	81.140	82.352	1.49%	3.59%	72.54%	23.86%	7.17%	74.16%	18.67%	20
		50	101.040	99.679	-1.35%	19.41%	58.98%	21.61%	23.31%	61.27%	15.42%	23
		100	132.188	133.222	0.78%	38.48%	45.06%	16.46%	42.62%	45.84%	11.54%	24
50	25	50	403.026	385.697	-4.30%	7.22%	80.77%	12.01%	9.80%	81.88%	8.32%	60
		100	454.968	441.124	-3.04%	17.82%	71.54%	10.64%	21.14%	71.59%	7.27%	64
		200	558.915	542.651	-2.91%	33.11%	58.23%	8.66%	35.89%	58.20%	5.91%	110
	50	50	398.710	386.463	-3.07%	1.41%	83.21%	15.37%	3.82%	84.60%	11.58%	120

Table 18. Comparison of capacitances for TSVs with wires above, below, and in the side of the TSVs under perfect TSV-to-TSV alignment. The computation time of our model is negligible for all the cases. W is the TSV width, S is the TSV-to-TSV spacing, H is the TSV height, and R is the runtime of Raphael in minutes.

TSV dimension (μm)			TSV capacitance (fF)			Breakdown of capacitive components							R
W	S	H	S_{\min}	Raphael	Our model	Error	Raphael			Our model			
							C_{TT}	C_{top}	C_{inter}	C_{TT}	C_{top}	C_{inter}	
5	5	5	0.5	8.055	8.572	6.03%	0.46%	40.28%	59.26%	0.13%	43.52%	56.35%	120
		20	1.0	16.280	15.570	-4.36%	50.92%	21.92%	27.16%	48.59%	20.11%	31.30%	120
		50	2.0	33.751	35.115	4.04%	81.32%	11.76%	6.92%	80.63%	10.72%	8.65%	120
		100	2.0	64.799	67.581	4.29%	90.27%	6.13%	3.60%	93.98%	5.23%	0.79%	143
10	5	10	1.0	24.174	24.981	3.23%	15.75%	52.61%	31.64%	16.41%	50.48%	33.11%	300

a 3×3 array and wires above and below the TSVs as shown in Figure 37 (f) and (a). The capacitance of the center TSV is computed and compared under the assumption that all other TSVs and wires are grounded. Table 17 shows capacitances for various TSV dimensions.

It is observed that the relative difference between Raphael and our modeling is less than 5.88% for all the cases and the average error is 2.51% which is very small. The breakdown of capacitive components is also shown in the table to show that our model for each capacitance component is accurate. In the table, C_{TT} is the TSV-to-TSV coupling

capacitance, C_{top} is the coupling capacitance between a TSV and wire pieces right above the TSV ($= C_{\text{top},1} + C_{\text{top},2}$), and C_{side} is the coupling capacitance between a TSV and wire pieces outside the top surface of the TSV ($= C_{\text{side},1} + C_{\text{side},2}$). The difference between Raphael simulation and our model is again very small, which shows that our model is highly accurate. Moreover, the result shows that C_{top} and C_{side} are not negligible when TSV is relatively short compared to the TSV width. Therefore, it is required to consider TSV-to-wire capacitance for the computation of TSV capacitance. Raphael runtime is much higher but the computation time of our model is negligible.

3.3.2 TSVs with Top, Bottom, and Side Neighbors

The second comparison is on a structure composed of TSVs with top, bottom, and side neighboring wires. The Raphael simulation structure for this comparison consists of nine TSVs forming a 3×3 array and wires above, below, and in the side of the TSVs as shown in Figure 37 (f) and Figure 38 (a). The capacitance of the center TSV is computed and compared under the assumption that all other TSVs and wires are grounded. Table 18 shows that the difference between Raphael simulation and our model is less than 6.03% for all cases and the average error is 4.39% which is acceptable for fast estimation of TSV capacitances. The breakdown of capacitive components also shows that our model is highly accurate in computing individual capacitive components as well. In Table 18, C_{inter} is the sum of $C_{\text{side},3}$, $C_{\text{side},4}$, and C_{bm} . Regarding computation time, Raphael runtime is excessively high since many wires exist in this simulation structure. Moreover, Raphael simulation could not be performed on more complicated structures due to its huge memory requirement (more than 6 to 8GB).

3.3.3 TSV under Misalignment

The third comparison is on TSVs under misalignment. The Raphael simulation structure contains nine TSVs on a 3×3 array and another 9 TSVs on top of these TSVs with misalignment. The simulated values are the capacitances of the center TSV. Table 19 shows the

Table 19. TSV capacitance under misalignment. M_{TSV} is the misalignment ratio. Capacitance values are reported in fF . The unit of width, spacing, and height is μm .

Width	Spacing	Height	M_{TSV}	Raphael	Our model	Error
5	5	5	0%	6.412	6.475	0.98%
			5%	6.269	6.449	2.87%
			10%	6.348	6.438	1.42%
			20%	6.504	6.469	-0.54%
		50	0%	64.117	64.753	0.99%
			5%	62.485	64.726	3.59%
			10%	62.642	64.716	3.31%
			20%	63.061	64.747	2.67%
20	20	20	0%	25.647	25.901	0.99%
			5%	25.078	25.795	2.86%
			10%	25.393	25.752	1.41%
			20%	26.017	25.876	-0.54%
		50	0%	64.117	64.753	0.99%
			5%	62.581	64.646	3.30%
			10%	62.898	64.604	2.71%
			20%	63.709	64.728	1.60%
50	50	100	0%	128.234	129.506	0.99%
			5%	125.201	129.239	3.23%
			10%	125.914	129.133	2.56%
			20%	128.303	129.443	0.89%

comparison. The result shows that the capacitance change due to misalignment is not significant if the misalignment ratio is less than 20%. The relative difference between Raphael simulation and our model is less than 3.59% for all the cases. If a rough approximation for misalignment is sufficient, C_{m2} can be neglected. However, including C_{m2} results in more accurate capacitance values.

3.3.4 Impact of TSV Capacitance on Delay

As TSV capacitance is not negligible, the impact of TSV capacitance on delay is presented in this experiment. Table 20 shows ratios of TSV capacitance to wire capacitance. When the wirelength (L) is short (up to $100\mu m$), TSV capacitance is much bigger than wire capacitance. For instance, the capacitance of a TSV whose width is $5\mu m$, spacing is $5\mu m$, and height is $50\mu m$ is $8.5\times$ bigger than the capacitance of a wire whose length is $50\mu m$. Similarly, TSV capacitance is $22.82\times$ bigger than wire capacitance when the TSV width is

Table 20. Comparison between TSV capacitance and wire capacitance. The ratio of TSV capacitance to wire capacitance is reported. Wire width is $0.2\mu\text{m}$, wire thickness is $0.36\mu\text{m}$, horizontal wire spacing is $0.2\mu\text{m}$, and vertical wire spacing is $0.3\mu\text{m}$. L is the wirelength.

TSV dimensions	$L(\mu\text{m})$	TSV height		
		$20\mu\text{m}$	$50\mu\text{m}$	$100\mu\text{m}$
width : $5\mu\text{m}$ spacing : $5\mu\text{m}$	50	4.37	8.50	15.38
	300	0.73	1.42	2.56
	1000	0.22	0.43	0.77
	2000	0.11	0.21	0.38
	5000	0.04	0.09	0.15
width : $20\mu\text{m}$ spacing : $20\mu\text{m}$	50	18.85	22.82	30.50
	300	3.14	3.80	5.08
	1000	0.94	1.14	1.52
	2000	0.47	0.57	0.76
	5000	0.19	0.23	0.30

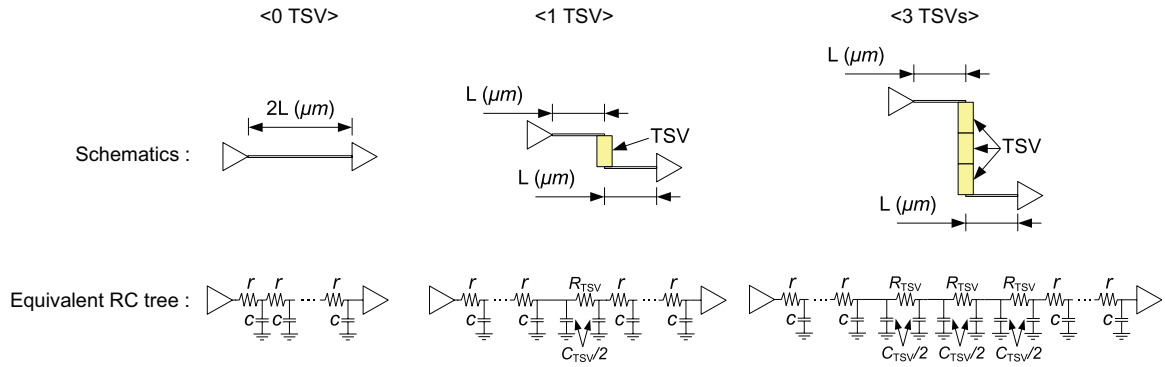


Figure 39. Schematics for the delay simulation in Table 21. C_{TSV} is the total capacitance (the sum of TSV-to-wire coupling capacitances and TSV-to-TSV coupling capacitances) of a TSV.

$20\mu\text{m}$, TSV-to-TSV spacing is $20\mu\text{m}$, the TSV height is $50\mu\text{m}$, and the wirelength is $50\mu\text{m}$. As the wirelength goes up, on the other hand, wire capacitance becomes much bigger than TSV capacitance.

Next, the impact of TSVs on 3D interconnect delay is presented. In our SPICE simulation, a signal goes through a wire, one TSV (or three TSVs), and then another wire whose length is same as that of the first wire as shown in Figure 39. Table 21 shows the delay values for various TSV dimensions. When the wirelength is short, the number of TSVs in the interconnect (1 vs 3 TSVs) affects the delay significantly. For instance, the delay of “3 TSVs” case is $2.06\times$ to $2.81\times$ bigger than “1 TSV” case when L is $50\mu\text{m}$. However,

Table 21. Delay of 3D interconnects. Schematics for this simulation are shown in Figure 39. All the delay values are scaled to the boldface case.

		TSV dimensions					
		$w = 5\mu m$ $s = 5\mu m$ $h = 20\mu m$		$w = 10\mu m$ $s = 10\mu m$ $h = 100\mu m$		$w = 20\mu m$ $s = 20\mu m$ $h = 50\mu m$	
$L(\mu m)$	0 TSV	1 TSV	3 TSVs	1 TSV	3 TSVs	1 TSV	3 TSVs
50	0.32	1.00	2.06	2.83	7.86	3.35	9.41
300	1.65	2.23	3.39	4.17	9.20	4.68	10.75
1000	5.39	5.98	7.14	7.92	12.97	8.44	14.53
2000	10.78	11.37	12.54	13.32	18.40	13.84	19.97
5000	27.31	27.91	29.10	29.90	35.06	30.42	36.65

Table 22. TSV-to-TSV coupling capacitance vs. TSV MOS capacitance. These numbers do not include TSV-to-wire capacitance. w is TSV width, h is TSV height, and s is TSV-to-TSV spacing.

TSV dimensions (in μm)			MOS cap. (fF)	Coupling cap. (fF)
w	h	s		
5	20	5	27.5	13.4
		10		8.78
	50	5	68.8	32.2
		10		21.2
10	50	10	74.3	32.2
		20		21.1
	100	10	148.5	63.6
		20		41.6

the impact of TSVs decreases as the wirelength increases because long wires have larger parasitic capacitance than TSVs so that wire capacitance becomes dominant in long wires.

3.3.5 Comparison Between TSV Coupling and MOS Capacitance

In previous work such as [30, 44], TSV MOS capacitance is used for TSV capacitance. Therefore, TSV coupling capacitance is compared to TSV MOS capacitance in this section. Table 22 compares the TSV-to-TSV coupling capacitance with TSV MOS capacitance computed by capacitance equations presented in [30] ⁴. It is observed from the table that the coupling capacitance is smaller than MOS capacitance. For example, when the TSV width is $10\mu m$ and the TSV height is $50\mu m$, MOS capacitance is $74.3fF$ but the coupling

⁴TSV-to-wire coupling capacitance is not included here.

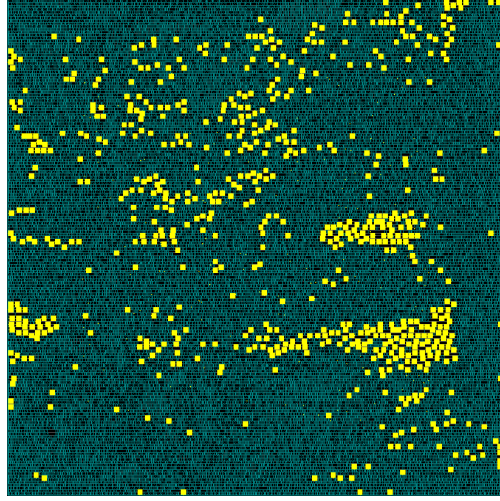


Figure 40. An example layout of a 3D IC designed by 3D IC design methodology presented in [11]. Via-first TSVs are used and two dies are stacked with face-to-back bonding. Bright rectangles are TSV landing pads (TSVs exist inside landing pads), and dark rectangles are standard cells.

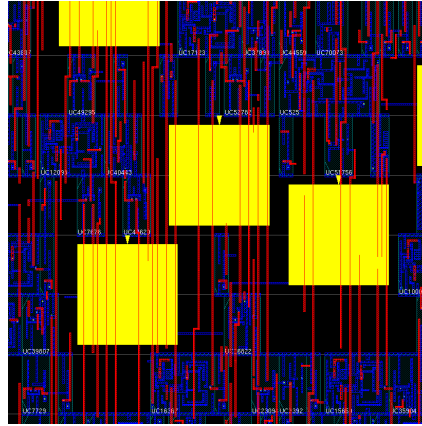


Figure 41. Zoom-in shot of Figure 40. Bright big rectangles are TSV landing pads (TSVs exist inside landing pads), and thin vertical lines above TSVs are metal wires.

capacitance is $21.1fF$ when the TSV-to-TSV spacing is $20\mu m$. The results indicate that using MOS capacitance is not accurate because it does not take TSV-to-TSV capacitive coupling into account.

3.4 Analyzing More General Layouts

The focus in previous sections is on two regular TSV structures, where a given TSV is surrounded by eight neighboring TSVs and full of wires above, below, and in the side of the TSVs. In real layouts, however, this kind of regular TSV arrangement rarely happens unless

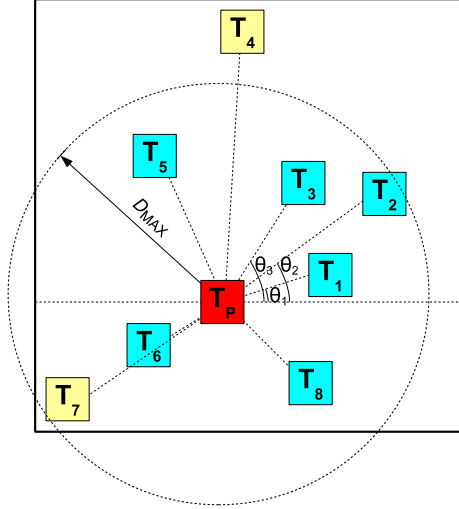


Figure 42. A general layout where TSVs are placed irregularly. The capacitance of T_P is computed.

highly regular TSV placement is used as presented in [11]. Figure 40 shows an example layout in which there are 152K cells and 428 TSVs, and Figure 41 shows a zoomed-in layout. It is observed that regular TSV structures do not occur in this layout. Rather, it is required to handle more general layouts. Therefore, a methodology for the analysis of general layouts is presented and its results are compared against Raphael simulation.

The first step used to compute TSV capacitance for irregularly-placed TSVs is to sort TSVs as follows. A horizontal line (l_0) is drawn so that it passes through the center of T_P , and a line (l_n) connecting the centers of T_P and T_n is also drawn as shown in Figure 42. The TSVs are sorted in the ascending order of the angle between l_0 and l_n , denoted by θ_n . The range of θ_n is greater than or equal to 0 (rad) and less than 2π (rad).

After sorting TSVs, *meaningful TSVs* for the given target TSV, T_P , are extracted. A *meaningful TSV* is a TSV, T_n , satisfying the following two conditions:

Distance condition : The distance from T_P to T_n is less than a distance D_{MAX} pre-determined empirically. For instance, the area capacitance between the facing sidewalls of two TSVs at a distance of d is approximately $1fF$, D_{MAX} is set to be d .

Visibility condition : T_n is visible from T_P . T_n is said to be visible from T_P if the

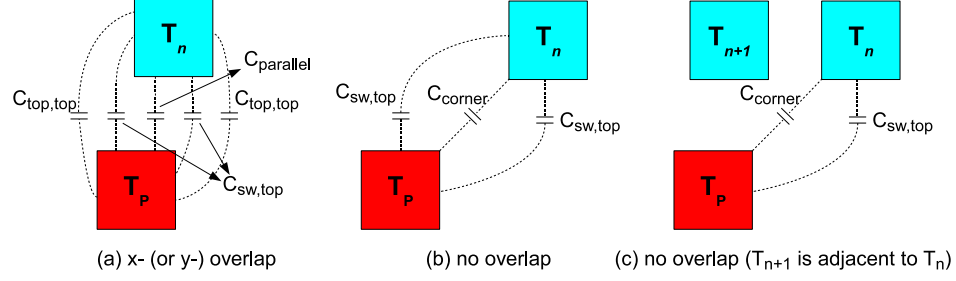


Figure 43. Capacitance computation for a pair of TSVs. If there exists an x - or y - overlap, $C_{parallel}$, $C_{sw,top}$, and $C_{top,top}$ are applied as shown in (a). If there is no overlap, C_{corner} and $C_{sw,top}$ are applied as shown in (b) and (c).

following inequality is satisfied:

$$|\theta_m - \theta_{m+1}| \geq \theta_{MIN} \quad (63)$$

where m is the TSV index, and θ_{MIN} is the pre-determined angle (0.1π in our simulations). If two adjacent TSVs in the sorted TSV list violate the visibility condition, the TSV having shorter distance from T_P is set to be a *meaningful TSV* and the other TSV is eliminated from the list. The sorted TSV list is circular. For instance, the angular difference between T_8 and T_1 in Figure 42 is computed to determine if one of them is a *meaningful TSV* or not.

TSVs that do not satisfy the distance condition are excluded during capacitance computation. The reason is that the coupling capacitance between T_P and T_n becomes too small if they are separated by a large distance. For instance, T_4 in Figure 42 is excluded due to violation of the distance condition.

TSVs that do not satisfy the visibility condition are also excluded during capacitance computation. They are excluded because electric field diverging from T_P does not reach T_n if another TSV exists in between T_P and T_n . For instance, T_7 in Figure 42 is excluded because $|\theta_6 - \theta_7|$ is less than θ_{MIN} and T_7 is farther away from T_P than T_6 .

3.4.1 Capacitance Computation for Meaningful TSVs

After the extraction of the list of meaningful TSVs, the capacitance of T_P is computed by summing the coupling capacitance between T_P and each meaningful TSV, T_n . The computation step is as follows. If there is an overlap in x - or y - coordinates of T_P and T_n

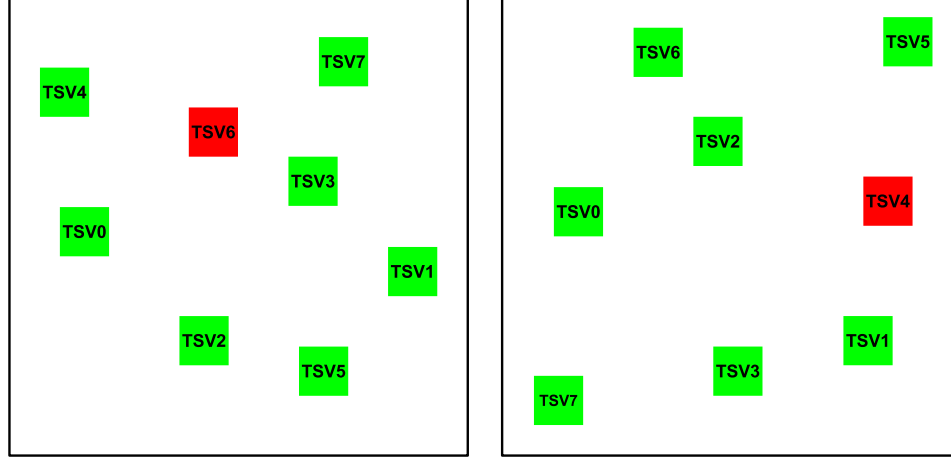


Figure 44. Two general (= non-regular) example layouts. The total number of TSVs is eight. The electric potential of one of them (= red square) is set to V_{DD} , while that of all others are set to 0.

as shown in Figure 43 (a), the parallel capacitance equation is applied to the overlapped region. For non-overlapped regions, $C_{sw,top}$ and $C_{top,top}$ are applied.

On the other hand, if there is no overlap in x - or y - coordinates of T_p and T_n as shown in Figure 43 (b), C_{corner} and $C_{sw,top}$ are applied. When $C_{sw,top}$ is applied, the relative position of T_n and T_{n+1} (or T_{n-1}) is also considered as shown in Figure 43 (c). If T_{n+1} (or T_{n-1}) blocks the path of the electrical field diverging from T_p to a sidewall of T_n , $C_{sw,top}$ is not applied for the sidewall of T_n .

3.4.2 Simulation Results

Simulation structures for the capacitance extraction on general layouts are constructed as follows. TSVs are first distributed in a fixed-size window as shown in Figure 44. Then a TSV among the distributed TSVs is chosen and its potential is set to be V_{DD} while the potentials of all other TSVs are set to be zero. For a randomly-generated layout, 1) the capacitance of the red TSV is obtained by our capacitance estimation program, 2) the structure is converted into Raphael input format, 3) the capacitance of the red TSV is computed by Raphael, and 4) the two capacitance values are compared. Figure 44 shows two example layouts. Each square represents a TSV, and the electric potential of green squares is set to zero while that of red square is set to V_{DD} .

Table 23. Capacitance extraction on general layouts.

TSV dimensions (μm)			# TSVs	Average error (%)	Max. error (%)
Width	Min. spacing	Height			
5	5	50	6	8.79	16.15
			8	10.20	15.59
			10	8.48	14.92
			12	11.86	15.79
		100	6	9.70	15.78
			8	10.49	16.94
			10	11.20	15.03
			12	8.53	14.82
10	10	50	6	10.68	16.15
			8	9.36	14.55
			10	11.44	17.18
			12	11.22	18.91
		100	6	10.10	17.06
			8	9.07	14.62
			10	10.08	15.48
			12	8.18	14.79

Table 23 shows the average relative errors between Raphael simulation and our model on random structures. For each simulation set (e.g., $5\mu m$ width and minimum spacing, $50\mu m$ height, and total six TSVs in the layout), 20 random structures are generated, errors for each structure are computed, and finally average and maximum errors out of 20 errors are obtained. In all the cases, the errors are less than 18.91% and the average error ranges between 8.18% and 11.86%, which is acceptable for fast estimation for quick full-chip timing analysis and layout optimization. The runtime of Raphael simulation is negligible when there are few objects and the layout boundary is small. However, it takes several seconds to compute coupling capacitances when there are more than ten objects and the layout boundary is large. Since this is the extraction runtime for one TSV, the actual runtime becomes N times longer when there exist N TSVs. On the other hand, our capacitance estimation is extremely fast. This clearly shows the effectiveness of our model for the fast estimation of TSV coupling capacitance.

3.5 Summary

The TSV coupling capacitance model in this chapter provides fast estimation of capacitance of TSVs surrounded by wires and other TSVs. The error between the model and Synopsys Raphael simulation on the two regular structures remains less than 6.03%, and the average error on more general structures is around 11.86%. However, this analytical model requires a fraction of Raphael simulation runtime to compute the coupling capacitance. Therefore, this fast and relatively accurate analytical model will enable more accurate TSV computation during design and optimization of 3D ICs.

CHAPTER 4

THE DESIGN OF GATE-LEVEL 3D INTEGRATED CIRCUITS

Three-dimensional integrated circuits (3D ICs) are emerging as a natural way to overcome interconnect scaling problems in 2D ICs. 3D ICs benefit from smaller footprint area than 2D ICs and from vertical (z-direction) interconnections between different dies [9, 33]. Small footprint area of 3D ICs allows gates to be placed closer, thereby leading to shorter wirelength than 2D ICs. Vertical interconnections by Through-Silicon-Vias (TSVs) also help shorten wirelength because gates can be placed on top of each other in different dies, eliminating the need of long cross-chip interconnects existing in 2D ICs. This shorter wirelength helps alleviate routing congestion as well as crosstalk and noise problems. Therefore, 3D ICs are expected to replace 2D ICs in the coming future.

Although TSVs can alleviate congestion, reduce wirelength, and improve performance, they occupy non-negligible silicon area. Excessive or ill-placed TSVs not only increase die area, but also have negative impact on these objectives in 3D ICs [33]. Therefore, CAD tools for 3D ICs should carefully consider the impact of TSVs during placement and routing. Depending on their type, via-first TSVs interfere with device layer, whereas via-last TSVs interfere with both device and metal layers (see Figure 45). A typical size of via-first TSVs ranges from $1\mu m$ to $5\mu m$, whereas that of via-last TSVs ranges from $5\mu m$ to $20\mu m$ [35]. These TSVs are much larger than wires, local vias, and gates, thus care must be taken to consider the impact of TSV usage on the layout of each die in a 3D stack. Most previous work on 3D IC CAD tools [26, 53], however, ignores either the sheer size of TSVs or the fact that TSVs interfere with gates and wires.

In this research, a new force-directed 3D placement algorithm is proposed. Two different TSV handling schemes, namely “TSV-site” and “TSV co-placement”, are also introduced. Since the TSV-site scheme requires assignment of 3D nets to TSVs, two TSV assignment algorithms are also developed. In addition, the placement and TSV assignment

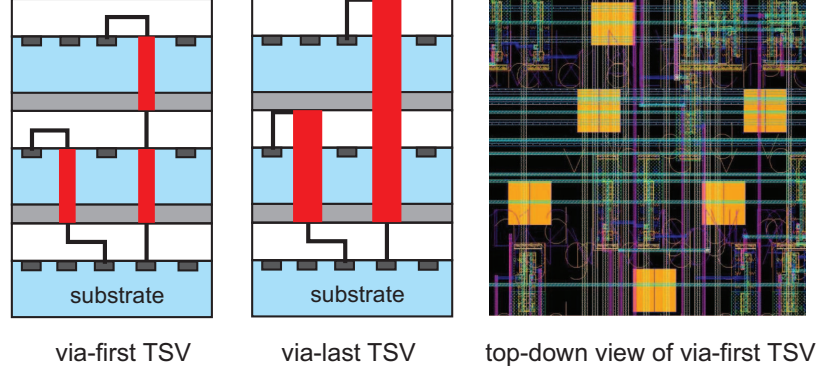


Figure 45. Via-first and via-last TSVs

algorithms are integrated into a commercial tool. This new tool flow generates GDSII-level 3D layouts that are fully validated. Based on these GDSII layouts, various studies on the impact of TSVs on 3D IC layouts are presented and demonstrated.

4.1 Preliminaries

4.1.1 Maximum Allowable TSV Count

TSVs occupy significant silicon area. However, previous researches on 3D placement and routing did not consider this fact. For example, the authors of [26] used 18,519 TSVs for ibm01 circuit which has 12,282 cells. If the average cell area is $2\mu m^2$ in 45nm technology, the total cell area becomes $24,564\mu m^2$. If a TSV occupies $10\mu m^2$, the total TSV area becomes $185,190\mu m^2$, which is $7\times$ bigger than the cell area. Similarly, the authors of [53] used about 15,000 TSVs for ibm01, which is not a realistic number.

Since the smallest 2D chip area is simply the total cell area, the maximum TSV count such that the chip area of a 3D IC is less than a pre-defined number can be computed easily. For instance, the maximum TSV count, $N_{TSV_{max}}$, based on 2D and 3D chip areas can be calculated by

$$N_{TSV_{max}} = (A_{3D} - A_{2D})/A_{TSV} , \quad (64)$$

where A_{3D} is the sum of the area of all dies in a 3D IC, A_{2D} is the area when the circuit is designed in 2D, and A_{TSV} is the area required by a TSV. If Equation (64) is applied to ibm01 in 45nm technology with $10\mu m^2$ TSV area, and $A_{3D} = 1.5 \times A_{2D}$, the maximum TSV

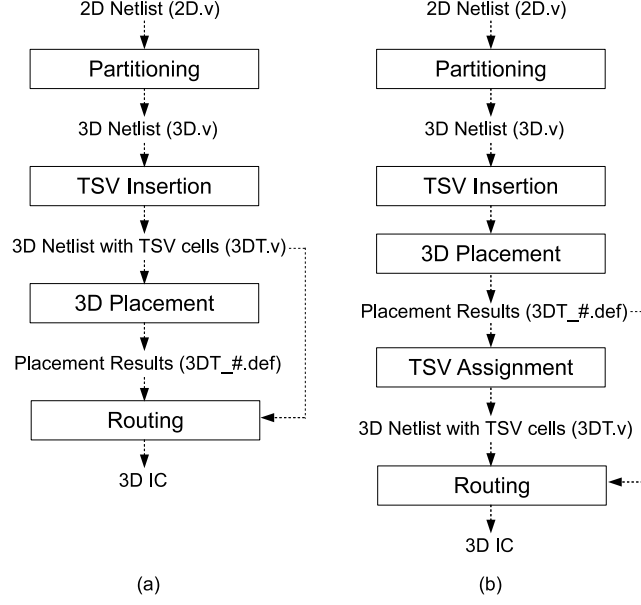


Figure 46. Two 3D IC design flows developed in this project. (a) TSV co-placement, (b) TSV-site

count is approximately 1,200. This result means that the total die area of the 3D IC will be greater than $1.5 \times A_{2D}$ if more than 1,200 TSVs are used.

4.1.2 Wirelength and TSV Count Trade-Off

TSVs help reduce wirelength because long wires in 2D ICs can be shortened by placing cells in a net on top of each other in different dies and connecting them with TSVs. However, TSVs have two negative impacts on the layout. First, they occupy silicon area, and interfere with cells, thereby spreading cells out so that the average distance between cells does not decrease as much as expected [33]. Second, TSVs contribute to routing congestion because they need to be connected to other cells. This impact becomes severe for via-last TSVs [22, 23] because these TSVs go through all metal layers (and device layers plus the bulk), and become routing obstacles. Therefore, designers have to wisely control the TSV usage [32]. In this project, the number of TSVs is controlled during partitioning.

4.1.3 3D IC Design Flow

Two 3D IC design flows, namely TSV co-placement and TSV-site, are devised in this project as shown in Figure 46. These flows are developed in such a way that existing 2D

routing tools can be re-used while TSVs are handled efficiently. The following shows the two 3D IC design flows.

Partitioning: In the first stage of both design schemes, cells in the 2D netlist are distributed into N_{die} dies by a modified FM partitioning. During the partitioning, the cutsize is controlled to obtain the desired number of TSVs. The output of this stage is the 3D netlist in which some of the 2D nets (nets having all their cells in a die) of the original design become 3D nets (nets having their cells in different dies). After partitioning is completed, the minimum number of TSVs to be inserted is computed. Although multiple TSVs can be used for a 3D net to connect cells in two adjacent dies, only one TSV is used for a 3D net between two adjacent dies.

TSV insertion and placement in TSV co-placement scheme: In TSV co-placement scheme, TSVs are added into the 3D netlist during TSV insertion stage, and then cells and TSVs are placed simultaneously during 3D placement. The output of the 3D placer is a DEF file for each die.

TSV insertion and placement in TSV-site scheme: In TSV-site scheme, TSVs are pre-placed uniformly on each die in TSV insertion stage, and then cells are placed in the 3D placement stage. During 3D placement, pre-placed TSVs are treated as placement obstacles because there should not be any overlap between a TSV and a cell. An additional stage, TSV assignment, is needed after 3D placement to determine which pre-placed TSVs belong to which 3D nets. Then, 3D netlists are updated to reflect the assigned TSVs.

Routing: After generating DEF and netlist files for each die, Cadence SoC Encounter [28] is used to route each die. Routing is done separately for each die because each die has its own netlist and cell positions. To facilitate TSV manipulation by Cadence SoC Encounter, a “TSV cell” is defined and used as if it is a standard cell.

4.2 3D Placement Algorithm

4.2.1 Overview of Force-Directed Placement

In quadratic placement, a placement result is computed by minimizing the quadratic wire-length function Γ , which can be expressed as

$$\Gamma = \Gamma_x + \Gamma_y, \quad (65)$$

where Γ_x and Γ_y are wirelength along x- and y-axis. Because Γ_x and Γ_y are independent, they can be separately minimized to obtain the minimum of Γ . The following description for x-dimension applies similarly to y-dimension. Here, Γ_x can be written in a matrix form as

$$\Gamma_x = \frac{1}{2} \mathbf{x}^T \mathbf{C}_x \mathbf{x} + \mathbf{x}^T \mathbf{d}_x + \text{constant}, \quad (66)$$

where $\mathbf{x} = [x_1 \cdots x_N]^T$ is a vector representing the x-position of N cells being placed, \mathbf{C}_x is an $N \times N$ matrix representing the connection among the cells along x-axis, and $\mathbf{d}_x = [d_{x,1} \cdots d_{x,N}]^T$ is a vector representing the connection to fixed pins along x-axis. Element $c_{x,ij}$ of \mathbf{C}_x matrix is the weight of connection between cell i and cell j , and element $d_{x,i}$ is the negative weighted position of fixed pins connected to cell i . The minimum of Γ_x can be obtained by setting its derivative to zero. Therefore, the cell placement along x-axis is computed by solving

$$\mathbf{C}_x \mathbf{x} + \mathbf{d}_x = \mathbf{0}. \quad (67)$$

Quadratic placement can be viewed as an elastic spring system when Γ is considered as the total spring energy of the system. Because the derivative of a spring energy is a force, the derivative of Γ_x in Equation (66) can be view as a net force $\mathbf{f}_x^{\text{net}}$ as

$$\mathbf{f}_x^{\text{net}} = \nabla_x \Gamma_x = \mathbf{C}_x \mathbf{x} + \mathbf{d}_x, \quad (68)$$

where $\nabla_x = [\partial/\partial_{x_1} \cdots \partial/\partial_{x_N}]^T$ is the vector differential operator. At equilibrium, $\mathbf{f}_x^{\text{net}}$ is zero, resulting in minimum Γ_x , but cells can be crowded in few area of the chip, resulting in high cell overlap.

Move force is density-based force that spreads cells away from high cell density area to low cell density area to reduce cell overlap. Move force in [54] is defined for 2D ICs. In 3D placement, move force is modified to support cell overlap removal in 3D ICs. Hold force is used to decouple each placement iteration from the previous iteration. It cancels out net force that pulls cells back to the placement in initial iteration, and can be written as

$$\mathbf{f}_x^{\text{hold}} = -(\mathbf{C}_x \mathbf{x}' + \mathbf{d}_x), \quad (69)$$

where $\mathbf{x}' = [x'_1 \cdots x'_N]^T$ is a vector representing the x-position of cells from the previous placement iteration. When no move force is applied, hold force holds cells being placed into their position.

Total force \mathbf{f}_x is the summation of net force, move force, and hold force. The total force is set to zero,

$$\mathbf{f}_x = \mathbf{f}_x^{\text{net}} + \mathbf{f}_x^{\text{move}} + \mathbf{f}_x^{\text{hold}} = \mathbf{0}, \quad (70)$$

to get the placement result with minimal wirelength and some cell overlap reduction for each placement iteration.

4.2.2 Overview of a 3D Placement Algorithm

Our 3D placement algorithm is divided into three phases: initial placement, global placement, and detail placement.

In the first phase, the initial placement is computed by solving Equation (67). The initial placement result contains high cell overlap, which will be reduced in each global placement iteration in the second phase by introducing move force and hold force in Equation (70), and solving the equation. Global placement continues until the amount of remaining cell overlap becomes low. Then, detailed placement starts in the third phase to legalize the result from global placement using a greedy algorithm.

4.2.3 Placing Cells in 3D ICs

It is not possible to extend the 2D force-directed quadratic placement algorithm to 3D placement algorithm simply by adding z-axis variable in Equation (65). The reason is that

all the fixed pins in 3D ICs are on the C4-bump side, resulting in placing all the cells at the same z-position, $\mathbf{z} = \mathbf{0}$, in the initial placement [53]. Therefore, the force-directed quadratic placement algorithm in [54] is extended by exploiting the fact that cells are already assigned into dies by the partitioner and not moving them across dies during placement.

Move force in [54] is modified to support placing cells in 3D ICs. Because cell overlap on all dies are different, move force for a cell is computed based on the cell overlap of the die on which the cell is being placed.

The placement problem is formulated as a global electrostatic problem by treating cell area as positive charge and chip area as negative charge. The placement density D on die d can be computed by

$$D(x, y)\Big|_{z=d} = D^{\text{cell}}(x, y)\Big|_{z=d} - D^{\text{chip}}(x, y)\Big|_{z=d}, \quad (71)$$

where $D^{\text{cell}}(x, y)\Big|_{z=d}$ is the cell density at position (x, y) computed by using only cells being placed on die d , and $D^{\text{chip}}(x, y)\Big|_{z=d}$ is the chip capacity scaled to match total area of cells being placed on the die.

After D is computed, placement potential Φ can be obtained by solving Poisson's equation

$$\Delta\Phi(x, y)\Big|_{z=d} = -D(x, y)\Big|_{z=d}. \quad (72)$$

The negative gradient of Φ indicates in which direction and how fast the cell at that position should move. Move force is modeled by connecting cell i to its target point \hat{x}_i with a spring of spring constant \hat{w}_i . The target point is computed by

$$\hat{x}_i = x'_i - \frac{\partial}{\partial x}\Phi(x, y)\Big|_{(x'_i, y'_i), z=d}, \quad (73)$$

where x'_i is the x-position of cell i being placed on die d from the previous placement iteration. Therefore, for cell i , move force $f_{x,i}^{\text{move}} = \hat{w}_i(x_i - \hat{x}_i)$, where x_i is the x-position of cell i being placed. Move force $\mathbf{f}_x^{\text{move}}$ is finally defined for 3D ICs by

$$\mathbf{f}_x^{\text{move}} = \hat{\mathbf{C}}_x(\mathbf{x} - \hat{\mathbf{x}}), \quad (74)$$

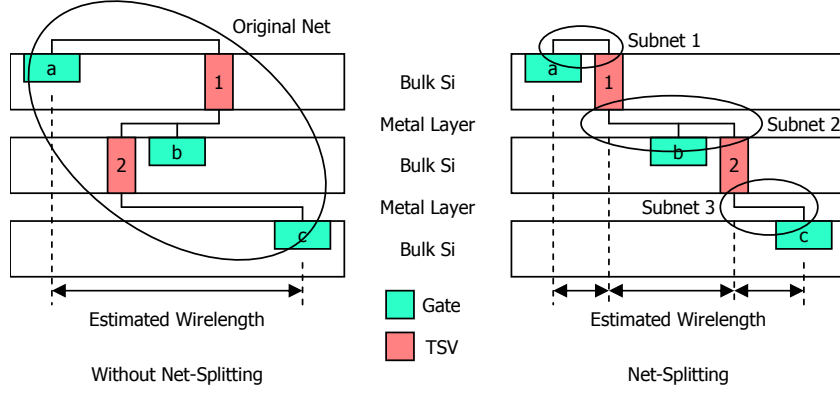


Figure 47. Splitting 3D net into subnets (side view)

where $\hat{\mathbf{C}}_x$ is a diagonal matrix of \hat{w}_i , $\mathbf{x} = [x_1 \cdots x_N]^T$ is a vector representing the x-position of N cells being placed, and $\hat{\mathbf{x}} = [\hat{x}_1 \cdots \hat{x}_N]^T$ is a vector representing the target x-position of the cells.

4.2.4 Placing TSVs in TSV Co-placement Scheme

In TSV co-placement scheme, a TSV is treated as a cell being placed. Therefore, our 3D placement algorithm is modified to place TSV cells in TSV co-placement scheme. After adding the minimum number of TSV cells into the netlist, the total number of cells being placed is updated. The area of TSV cells is also used to compute $D^{\text{cell}}(x, y)|_{z=d}$ and $D^{\text{chip}}(x, y)|_{z=d}$ in Equation (71). The resulting \mathbf{x} vector obtained from solving Equation (67) and (70) also includes the x-position of TSV cells.

4.2.5 Net Splitting

During wirelength computation, net splitting is used to compute wirelength more accurately as shown in Figure 47. Wirelength computation without net splitting is based on the projection of the cell locations in all dies onto a 2D plane. On the other hand, wirelength computation with net splitting is based on the projection of the cell locations in each die onto its own 2D plane. Therefore net splitting during wirelength computation gives us more accurate wirelength estimation.

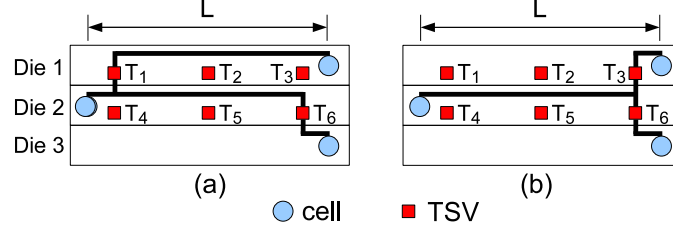


Figure 48. Cost computation for each combination of TSVs in three dies (side view). (a) wirelength = $2L$ for (T_1, T_6) , (b) wirelength = L for (T_3, T_6)

4.2.6 Pre-placing TSVs in TSV-site Scheme

In TSV-site scheme, TSVs are pre-placed into placement area before the original cells are placed. Therefore, pre-placed TSVs are treated as placement obstacles. Although the total number of cells being placed is not updated, and the resulting \mathbf{x} vector obtained from solving Equation (67) and Equation (70) still includes only the x-position of the original cells in the design. Therefore, the area of pre-placed TSVs is included when computing $D^{\text{cell}}(x, y)|_{z=d}$ and $D^{\text{chip}}(x, y)|_{z=d}$ in Equation (71).

TSVs are evenly pre-placed as placement obstacles in rows and columns in this scheme. Placement obstacles can be handled naturally by the mean of placement density in [54]. By including the area of pre-placed TSVs when computing placement density, move force is altered in such a way that it drives cells being placed away from pre-placed TSVs.

4.3 TSV Assignment

TSV assignment problem is to assign 3D nets to TSVs for given sets of dies, 3D nets, placed cells, and placed TSVs so that the total wirelength of 3D nets is minimized. The constraints are: (1) a TSV cannot be assigned to more than one 3D net, and (2) a 3D net should use one TSV between two adjacent dies.

4.3.1 Optimum Solution for TSV Assignment

A Binary Integer Linear Programming (BILP) formulation to find the optimum solution of TSV assignment for two dies was already shown in [55]. Since the number of binary integer variables in the formula was too big, the authors in [55] introduced heuristic algorithms

based on neighborhood search.

If more than two dies exist and a 3D net spans in more than two dies, combinations of TSVs in different dies should be considered for cost computation (see Figure 48). In Figure 48(a), T_1 and T_6 are assigned to the 3D net, and the cost (=wirelength) is approximately $2L$. On the other hand, T_3 and T_6 are assigned to the 3D net in Figure 48(b), and the cost is approximately L . Although T_6 is used in both cases, its contributions to the cost are different. Therefore, the cost should be computed for each combination of TSVs.

The optimum solution of TSV assignment for the case of more than two dies is found by the following formulation:

Minimize

$$\sum_{i=1}^{N_{3DNet}} \sum_{k=1}^{CB_i} \sum_{p=1}^{N_{TSV}} d_{i,k,p} \cdot x_{i,k,p} \quad (75)$$

Subject to

$$\sum_{k=1}^{CB_i} \sum_{p=1}^{N_{TSV}} x_{i,k,p} = 1, \quad (i = 1, \dots, N_{3DNet}) \quad (76)$$

$$\sum_{i=1}^{N_{3DNet}} \sum_{k=1}^{CB_i} x_{i,k,p} \leq 1 \quad (p = 1, \dots, N_{TSV}) \quad (77)$$

where N_{3DNet} is the total number of 3D nets, N_{TSV} is the total number of TSVs, CB_i is the total number of combinations of TSVs for the 3D net H_i , and $d_{i,k,p}$ is the cost when the k -th combination is used for the 3D net H_i . Here, $x_{i,k,p}$ is 1 if (1) the 3D net H_i uses the combination CB_k , and (2) the combination CB_k uses the TSV T_p , and otherwise $x_{i,k,p}$ is 0. Equation (76) denotes that a 3D net uses only one combination, and Equation (77) denotes that a TSV is assigned to at most one 3D net.

The number of variables in this problem is also very big because all possible combinations for all 3D nets should be considered. In addition, the number of combinations is still big even when available TSVs for a 3D net are limited to TSVs inside a small window. For example, if a 3D net spans in four dies, and the window contains 20 TSVs in each die, 8,000 combinations are available for the net. Moreover, limiting the window size may

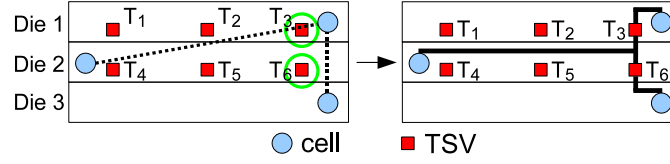


Figure 49. MST-based TSV assignment (side view)

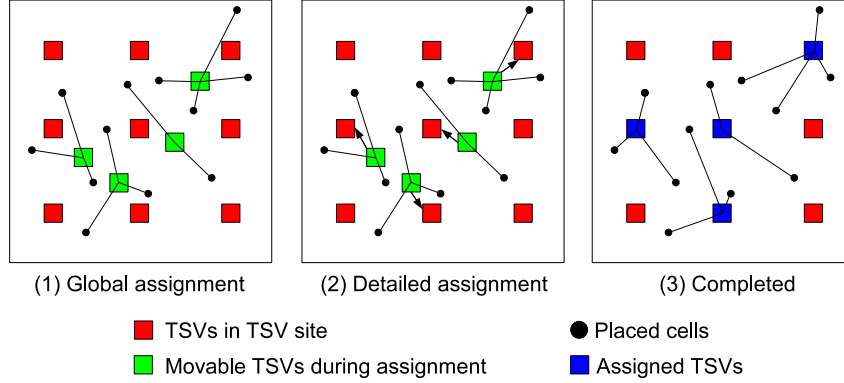


Figure 50. TSV assignment based on 3D Placement (top view)

result in the infeasibility of BILP. Therefore, two heuristic algorithms are introduced in this project.

4.3.2 MST-based TSV Assignment

In this method, minimum spanning tree (MST) is used for TSV assignment as shown in Figure 49. After constructing MST for a 3D net, the nearest TSV to the shortest edge is selected. Then, this process is iterated for the next shortest edge until all the dies are connected by TSVs. In Figure 49, the shortest edge spans in all the three dies so that the nearest TSV in each die is assigned to the edge.

MST-based TSV assignment is a sequential (net by net) method. Therefore, the order of nets for assignment becomes important because 3D nets assigned at the beginning have more available TSVs. In our method, 3D nets are sorted in the ascending order of bounding-box size because a net which has a large bounding box containing many TSVs inside has more choices for its TSVs.

Table 24. Benchmark Circuits

Circuit	# gates	# TRs	# nets	Profile
Ind 1	16K	137K	12K	Microprocessor
Ind 2	15K	106K	15K	Inverse DCT
Ind 3	16K	134K	16K	Microprocessor
Ind 4	20K	146K	20K	Microprocessor
Ind 5	30K	317K	30K	Arithmetic Unit
ethernet	77K	729K	77K	Ethernet IP Core
RISC	88K	775K	89K	Microprocessor
b18	104K	728K	104K	Microprocessor Cores
des_perf	109K	823K	109K	DES (Data Encryption Standard)
b19	169K	1.29M	169K	Microprocessor Cores

Table 25. Wirelength of our 3D placement with and without net-splitting

Circuit	without net-splitting (μm)	with net-splitting (μm)	Dif.(%)
Ind 1	444,867	408,713	-8.13%
Ind 2	309,936	288,143	-7.03%
Ind 3	305,961	308,006	+0.67%
Ind 4	405,010	393,215	-2.91%
Ind 5	658,886	584,024	-11.36%
ethernet	1,538,792	1,406,073	-8.62%
RISC	2,225,730	2,025,187	-9.01%
b18	2,610,358	2,683,424	+2.80%
des_perf	2,362,977	2,199,149	-6.93%
b19	4,612,405	4,364,694	-5.37%
Average			-5.59%

4.3.3 Placement-based TSV Assignment

In this method, the assignment problem is solved by 3D placement algorithm. The placed cells, however, become fixed cells at this time, and TSVs become movable cells. The assignment is done in two steps - global and detailed. Figure 50 shows how TSVs are assigned by 3D placement. The global assignment is done by 3D global placement. During this step, TSVs are placed by force-directed quadratic method regardless of TSV-site locations. After global placement is done, the detailed assignment is performed by cell snapping. In this step, each TSV is assigned to each TSV-site.

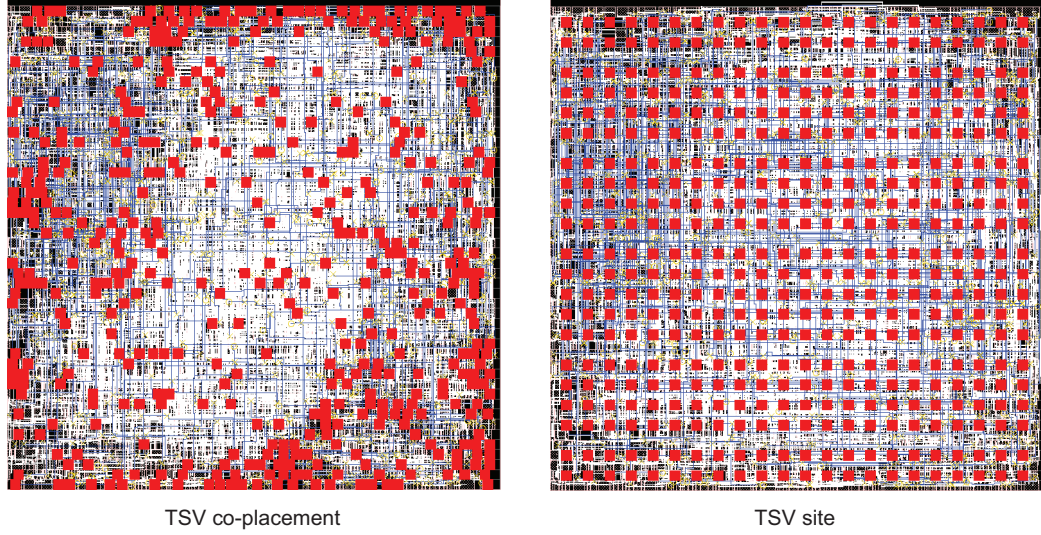


Figure 51. Cadence SoC Encounter snapshot of the bottommost die of Ind2 designed by TSV co-placement and TSV-site methods. Routing for 3D nets are shown in blue.

Table 26. Comparison of wirelength (WL), the minimum number of metal layers (ML), runtime for placement, and total silicon area for 2D and 3D (4 dies) design for IWLS 2005 benchmarks and industrial circuits. Cell occupancy is 80%, and the number of 3D nets was set to be 3% to 5% of the number of total nets during partitioning. The numbers in parentheses are ratios to 2D.

Circuit	our 2D				our 3D				
	WL (μm)	# ML	runtime (s)	Area (μm^2)	WL (μm)	# ML	runtime (s)	Area (μm^2)	# TSVs
Ind 1	397,015 (1.0)	5	85 (1.0)	44,944 (1.0)	399,924 (1.01)	4	93 (1.10)	69,696 (1.55)	1,700
Ind 2	334,648 (1.0)	4	72 (1.0)	44,944 (1.0)	284,340 (0.85)	4	53 (0.73)	58,564 (1.30)	1,302
Ind 3	287,587 (1.0)	4	71 (1.0)	48,841 (1.0)	300,781 (1.05)	4	81 (1.14)	69,696 (1.43)	798
Ind 4	411,993 (1.0)	4	157 (1.0)	63,001 (1.0)	388,315 (0.94)	4	101 (0.64)	80,656 (1.28)	1,016
Ind 5	703,461 (1.0)	5	189 (1.0)	103,684 (1.0)	582,603 (0.83)	4	188 (1.00)	147,456 (1.42)	2,789
ethernet	1,534,386 (1.0)	4	1,289 (1.0)	293,764 (1.0)	1,401,059 (0.91)	4	1,287 (1.00)	341,056 (1.16)	3,866
RISC	1,976,549 (1.0)	4	880 (1.0)	314,721 (1.0)	2,001,986 (1.01)	4	727 (0.83)	386,884 (1.23)	4,438
b18	2,415,867 (1.0)	5	1,459 (1.0)	338,724 (1.0)	2,683,424 (1.11)	4	1,134 (0.78)	495,616 (1.46)	10,404
des.perf	2,445,398 (1.0)	5	1,367 (1.0)	327,184 (1.0)	1,911,731 (0.78)	4	950 (0.69)	386,884 (1.18)	3,856
b19	3,986,586 (1.0)	5	2,642 (1.0)	580,644 (1.0)	3,945,515 (0.99)	4	2,173 (0.82)	712,336 (1.23)	8,497

4.4 Simulation Results

IWLS 2005 benchmarks [29] and several industrial circuits are used for 3D placement. They are listed in Table 24. 45nm technology is also used for experiments. TSV cell size is $2.47\mu m \times 2.47\mu m$.

4.4.1 Net-splitting Results

The first experiment is on the effectiveness of net-splitting for wirelength computation. Table 25 shows the wirelength comparison. Although “without net-splitting” is better for two circuits, “with net-splitting” is generally better, and the average improvement is 5.59%. The reason that “with net-splitting” generates shorter wirelength is that it estimates wirelength

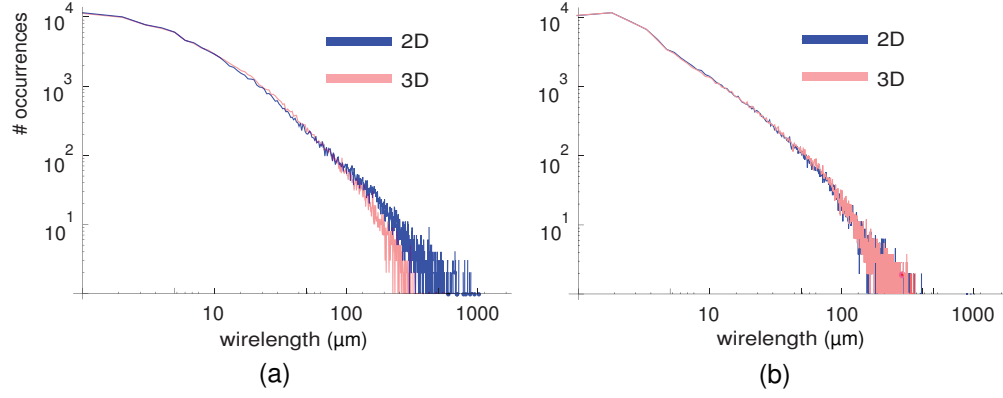


Figure 52. Wirelength distribution of (a) *des_perf*, where the die width is $572\mu\text{m}$ in 2D design and $311\mu\text{m}$ in 3D design (4 dies), (b) *b19*, where the die width is $762\mu\text{m}$ in 2D design and $411\mu\text{m}$ in 3D design (4 dies).

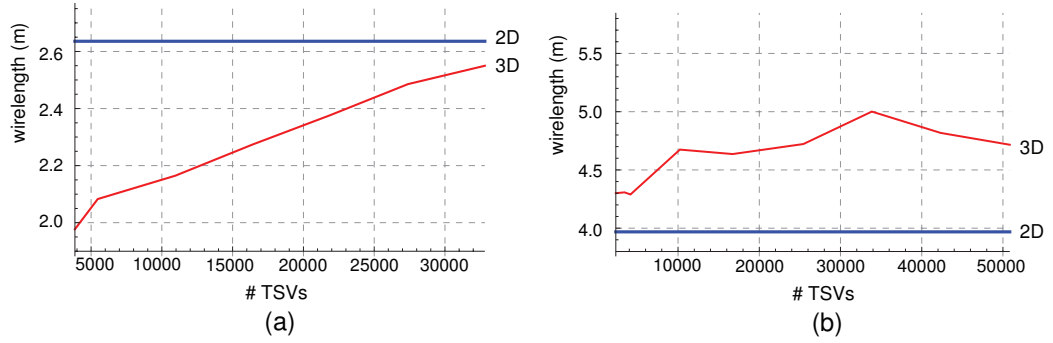


Figure 53. Wirelength vs # TSVs of (a) *des_perf*, and (b) *b19* for 2D and 3D (4 dies) designs

more accurately in a 3D view so that it makes our placer reduce the total wirelength more effectively. For the rest of the experiments, therefore, net-splitting is used for wirelength estimation.

4.4.2 Wirelength and Runtime Comparison

The second experiment is on the comparison of wirelength and runtime of 2D and 3D placement algorithms. Table 26 shows wirelength and runtime of our 2D placement and 3D placement results. The wirelength reduction in non-microprocessor circuits is 10% to 20% in 3D. However, it was not possible to benefit from 3D design in terms of wirelength for microprocessors.

To figure out the reasons, wirelength distributions are shown in Figure 52 for *des_perf* which is a non-microprocessor circuit, and for *b19* which is a set of microprocessors. As

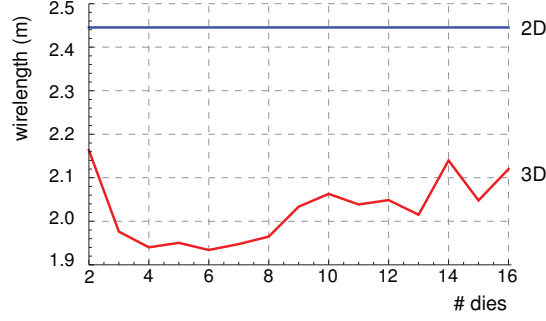


Figure 54. Wirelength vs # dies of des_perf in 3D design

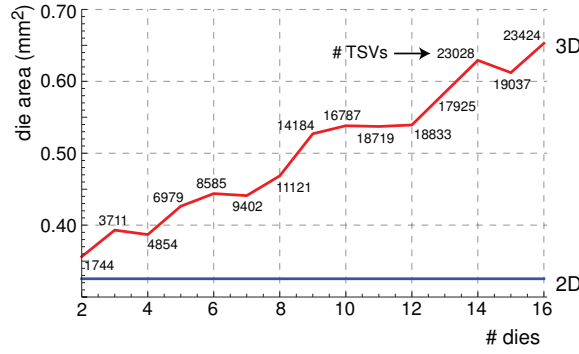


Figure 55. Die area and # TSVs of des_perf in 3D design

shown in Figure 52, long interconnections of des_perf in 2D become shorter in 3D. The longest wire of des_perf in 2D design is about $1000\mu\text{m}$ -long, whereas the longest wire in 3D design is about $320\mu\text{m}$ -long. This effect obviously comes from smaller footprint area than 2D design and connections in z-direction.

On the other hand, long interconnections of b19 in 2D do not become shorter in 3D. Since partitioning is used as a pre-process for 3D placement, the min-cut 4-way partitioning results show that the cut size of des_perf is 1,613(1.47%) out of 109,415 nets, whereas the cut size of b19 is 253(0.15%) out of 169,470 nets. This cut size means that b19 is highly modularized so that the total wirelength cannot be reduced much if min-cut partitioning is used.

Runtime of 3D placement is smaller than 2D placement. The reason is that 3D placement results have smaller number of overlaps than 2D placement results because each die in 3D ICs has less number of cells to be placed. Since force-directed quadratic placement

Table 27. Comparison of wirelength of TSV co-placement, TSV-site placement with MST-based TSV assignment, and TSV-site placement with placement-based TSV assignment. The numbers in the parentheses are ratios to TSV co-placement.

Circuit	Wirelength (μm)		
	TSV co-placement	TSV-site	
		MST-based	Placement-based
Ind 2	284,340 (1.0)	310,677 (1.09)	312,423 (1.10)
ethernet	1,401,059 (1.0)	1,513,381 (1.08)	1,554,960 (1.11)
des_perf	1,911,731 (1.0)	2,197,209 (1.15)	2,228,375 (1.17)
Runtime for assignment (s)			
Ind 2	-	0.08	34
ethernet	-	2.86	188
des_perf	-	1.13	290

algorithm spends a significant portion of its runtime in removing overlaps, having less number of cells in a die improves runtime.

4.4.3 Metal Layers and Silicon Area Results

The third experiment is on the number of metal layers and silicon area. Since 3D design has smaller footprint area than 2D design, and each die has less number of cells, the number of metal layers required for 3D design could be smaller than that for 2D design. Table 26 shows the comparisons of the minimum number of metal layers in 2D and 3D designs. While all the circuits are routable with 4 metal layers in 3D designs, some of the 2D designs are not routable with 4 metal layers because of congestion (DRC errors). The benefit of the decreased number of metal layers in 3D design comes from TSV insertion which results in the increase of the silicon area. Table 26 also shows how much area in 3D design increased.

4.4.4 On Wirelength vs # TSVs

The fourth experiment is on relationships between wirelength and the number of TSVs. Figure 53 shows the results for des_perf and b19. The wirelength of des_perf in 3D design monotonically increases as the TSV count increases. This result indicates that the additional TSVs do not help wirelength reduction much. They rather increase die area thereby increasing the wirelength. On the other hand, the wirelength of b19 in 3D design generally

increases at first as the TSV count increases, but it saturates after all. From this, it is observed that using too many TSVs will eventually increase the die area, which will result in wirelength increase.

4.4.5 On Wirelength and Die Area vs # Dies

The fifth experiment is on relationships of wirelength, die area, and the number of dies. In this experiment, the number of dies (N_{die}) is varied from 2 to 16, and wirelength, die area, and the number of TSVs are recorded. The wirelength of des_perf in 3D design dramatically decreases as N_{die} increases up to 4, then it saturates or slightly goes up as shown in Figure 54. If N_{die} increases more, the TSV count and die area will go up as shown in Figure 55. In other words, increasing N_{die} is helpful at first, but becomes not helpful as N_{die} goes up because 1) the TSV count increases, 2) the increased TSV count leads to the increase of die area, and 3) some of the 2D nets do not need to be 3D nets. This trend may not be applicable to all the 3D designs. However, using a small number of TSVs is helpful if partitioning is used as a pre-process for 3D placement.

4.4.6 TSV Co-placement vs TSV-site

The final experiment is on the comparison of TSV co-placement and TSV-site. Table 27 shows wirelengths of TSV co-placement and TSV-site. The wirelength increase of TSV-site placement with MST-based TSV assignment compared to TSV co-placement is 8% to 15%, whereas the wirelength increase of TSV-site placement with placement-based TSV assignment is 10% to 17%. Runtime overhead is a few seconds for MST-based TSV assignment and a few minutes for placement-based TSV assignment. Although TSV co-placement was better than TSV-site with respect to wirelength, TSV-site has its own advantages which are “better heat dissipation and stronger package bonding” according to [55].

4.5 Summary

In this chapter, two 3D IC design flows, TSV co-placement and TSV site, are proposed. In the TSV co-placement design scheme, gates and TSVs are placed simultaneously. In the TSV site design scheme, on the other hand, TSVs are uniformly placed and then gates are placed while the pre-placed TSVs are treated as obstacles. The TSV assignment step, which assigns 3D nets to pre-placed TSVs, follows the gate placement in the TSV site design scheme. For 3D placement, an existing force-directed 2D placement algorithm is extended to 3D. Two TSV assignment algorithms, 3D MST-based and 3D placement-based algorithms, are also developed for the TSV site design scheme. The simulation results show that the proposed design methodologies and algorithms effectively insert and place TSVs. 3D IC layouts generated by the proposed algorithms have shorter wirelength and use fewer metal layers although the die area slightly increases because of TSV insertion. Timing and power of 3D designs are worse than those of 2D designs for small circuits, but the opposite results are observed for large circuits.

CHAPTER 5

THE DESIGN OF BLOCK-LEVEL 3D INTEGRATED CIRCUITS

As 2D ICs are designed at various design levels such as block level and gate level, 3D ICs can also be designed at various design levels. In the core-level 3D IC design, existing 2D IC layouts are put together, signal, power/ground, thermal, and dummy TSVs are inserted, and each die is fabricated, stacked, and bonded. The primary merit of the core-level design is that 2D CAD tools can be fully utilized to design each die and reuse highly-optimized 2D IC layouts.

In the block-level 3D IC design, 3D floorplanning is performed with existing 2D blocks, TSVs are inserted into whitespace, and dies are fabricated, stacked, and bonded. The primary merit of the block-level design is that existing highly-optimized blocks can be reused without major modification. Since re-designing and re-optimizing each block in a 3D fashion is very costly, using existing well-designed blocks is inevitable in the 3D IC design.

In the gate-level 3D IC design, the whole design is flattened, gates and TSVs are placed in 3D, and dies are fabricated, stacked, and bonded. Since the gate-level 3D IC design provides the highest degree of freedom on gate and TSV locations, previous works focus on the gate-level 3D IC design. However, re-designing a whole circuit in the gate-level 3D IC design significantly increases design cost. In addition, pre-bond testing is also becoming a serious overhead in this design level [56].

One of the most important issues in the 3D IC design is that locations of signal TSVs have a huge impact on the design quality. Ill-placed signal TSVs cause long detours, so the performance of 3D ICs having poorly-placed TSVs could be worse than that of 2D ICs. Therefore, signal TSV locations should be taken into account in the 3D IC design. While many papers address signal TSV insertion in the core-level and the gate-level 3D

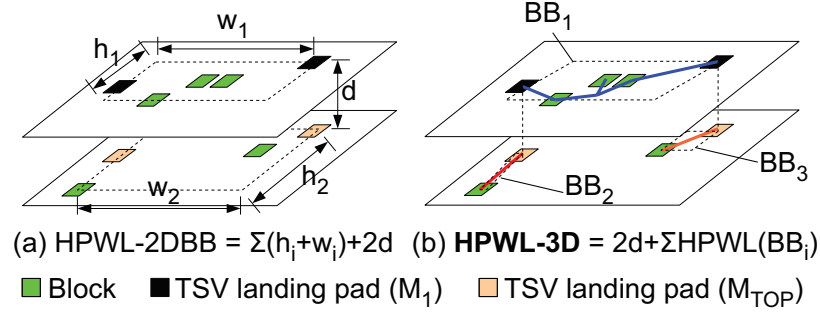


Figure 56. Wirelength metrics for a 3D net. (a) HPWL based on 2D bounding boxes. (b) HPWL based on subnet construction. d is the vertical length of a TSV.

ICs [11, 57, 1, 58], few work inserts signal TSVs physically in the block-level 3D IC design [59, 3, 60]. In addition, some of these block-level 3D IC design works do not use realistic wirelength metrics, so they significantly underestimate total wirelength. Furthermore, they do not consider multiple signal TSV insertion, which is essential for wirelength minimization. In this research, therefore, design methodologies and algorithms are developed for signal TSV planning in the block-level 3D IC design.

5.1 3D Wirelength Metrics

In this section, 3D wirelength metrics are reviewed and a more accurate wirelength metric is proposed for use in the multiple TSV insertion. The following terminologies distinguish two signal TSV insertion methods.

- **Single TSV insertion:** Only one TSV is inserted to connect blocks placed in two adjacent dies.
- **Multiple TSV insertion:** Multiple TSVs are inserted to connect blocks placed in two adjacent dies if inserting multiple TSVs reduces the total wirelength further.

5.1.1 3D Half-Perimeter Wirelength Based on Bounding Boxes

One simple way to compute the wirelength of a 3D net is to construct a *3D bounding box* containing blocks and TSVs in the 3D net and sum the width, the height, and the vertical length of the 3D bounding box. This wirelength metric is called *HPWL-3DBB* (HPWL

based on a 3D bounding box). [59, 3] use this wirelength metric. However, HPWL-3DBB significantly underestimates the wirelength.

Another way to compute the wirelength of a 3D net is to construct *2D bounding boxes* containing blocks and TSVs in each die in the 3D net. After 2D bounding box construction in each die, the HPWL of each 2D bounding box and the vertical length of a TSV multiplied by the number of TSVs are summed. This wirelength metric is called *HPWL-2DBB* (HPWL based on 2D bounding boxes). Figure 56(a) shows an example of HPWL-2DBB. If the single TSV insertion is used, HPWL-2DBB produces the most accurate HPWL-based 3D wirelength.

5.1.2 Subnet-based 3D Half-Perimeter Wirelength

If the multiple TSV insertion is used, HPWL-2DBB computes the wirelength of a 3D net inaccurately. In fact, the multiple TSV insertion splits a 3D net into multiple subnets as shown in Figure 56(b). In this case, each subnet has its own bounding box, so the total wirelength of a 3D net H_i can be computed more accurately as follows:

$$\text{HPWL-3D}(H_i) = d \cdot N_{\text{TSV},i} + \sum \text{HPWL}(\text{BB}_{i,j}) , \quad (78)$$

where d is the vertical length of a TSV, $N_{\text{TSV},i}$ is the total number of TSVs used for net H_i , and $\text{HPWL}(\text{BB}_{i,j})$ is the HPWL of the 2D bounding box of the j -th subnet of H_i . HPWL-3D also computes the wirelength of the single TSV insertion accurately.

5.2 Signal TSV Planning

Figure 57 shows the proposed signal TSV planning flow for the block-level 3D IC design. It is assumed that 3D floorplans are given. For a given 3D floorplan, TSV locations minimizing wirelength are found regardless of locations of available whitespace. To find TSV locations, a 3D rectilinear Steiner tree (RST) is constructed for each 3D net, a bottom-up breadth-first search is applied to the 3D RST to find a die span of each Steiner point, and TSV locations are determined.

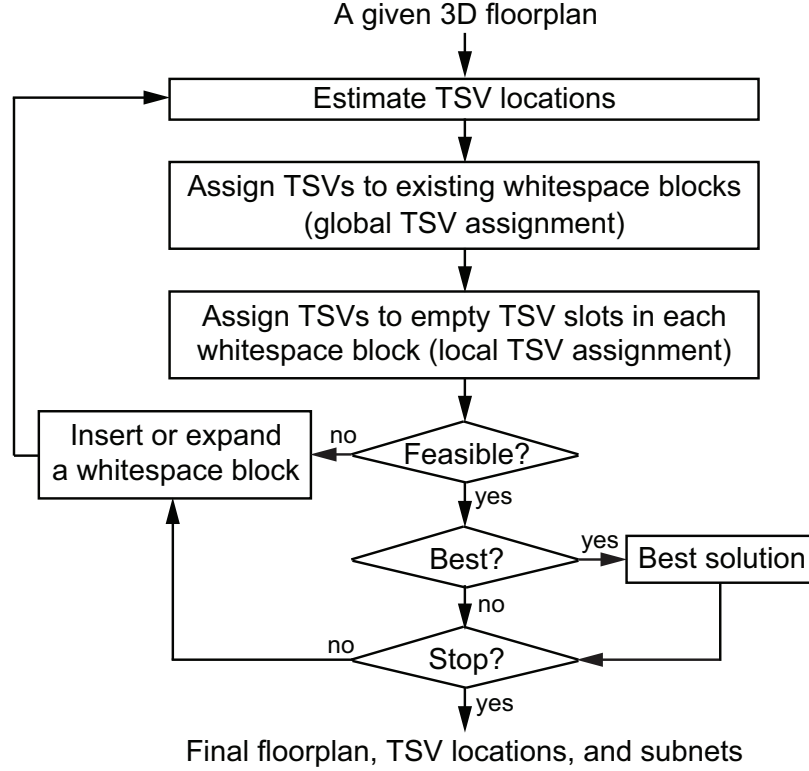


Figure 57. The proposed signal TSV planning flow.

In general, floorplanners generate compact floorplans, so TSV locations found by algorithms ignoring available whitespace locations are likely to be located on functional blocks. Since TSVs cannot be inserted into functional blocks, available whitespace close to estimated TSV locations are found. This problem is solved by TSV assignment.

If assigning TSVs to whitespace fails because of lack of enough whitespace, a new whitespace block is inserted or an existing whitespace block is expanded or whitespace blocks are redistributed. Since this whitespace manipulation changes the given floorplan, estimation of TSV locations shows be performed again as shown in Figure 57. It is also assumed that via-first TSVs and face-to-back die stacking are used as illustrated in Figure 58.

5.3 Estimation of TSV Locations

2D rectilinear Steiner minimum tree (RSMT) construction algorithms are frequently used to find optimal routing topologies for 2D nets. Similarly, since a planar (x- or y-directional)

Algorithm 1: The 3D RST construction algorithm.

Input: A set $F = \{p \mid p \in \mathbb{Z}^3\}$ of fixed 3D points.
Output: TSV locations and subnets.

```
1  $E \leftarrow \text{Construct 2D RSMT}(F)$ ;  
2  $Q \leftarrow \{\}$ ; // a queue.  
3 for each  $p \in F$  do  
4    $p.\text{visited} \leftarrow \text{true}$ ;  
5    $p.\text{top} \leftarrow p.\text{die}$ ;  $p.\text{bot} \leftarrow p.\text{die}$ ;  
6    $Q.\text{enqueue}(p)$ ;  
7 end  
8 while  $!Q.\text{empty}()$  do  
9    $p_1 \leftarrow Q.\text{dequeue}()$ ;  
10  for each unvisited point  $p_2$  adjacent to  $p_1$  do  
11     $t\text{Top} \leftarrow \infty$ ;  $t\text{Bot} \leftarrow -\infty$ ;  
12    for each visited point  $p_3$  adjacent to  $p_2$  do  
13       $t\text{Top} \leftarrow \text{MIN}(p_3.\text{bot}, t\text{Top})$ ;  
14       $t\text{Bot} \leftarrow \text{MAX}(p_3.\text{top}, t\text{Bot})$ ;  
15    end  
16    if  $t\text{Top} > t\text{Bot}$  then  
17       $t\text{Top} \leftarrow \text{IRand}(t\text{Bot}, t\text{Top})$ ;  
18       $t\text{Bot} \leftarrow t\text{Top}$ ;  
19    end  
20     $p_2.\text{top} \leftarrow t\text{Top}$ ;  $p_2.\text{bot} \leftarrow t\text{Bot}$ ;  
21     $p_2.\text{visited} \leftarrow \text{true}$ ;  
22    for each unvisited point  $p_3$  adjacent to  $p_2$  s.t.  $p_3 \notin Q$  do  
23       $Q.\text{enqueue}(p_3)$ ;  
24    end  
25  end  
26 end
```

After 3D RST construction, TSVs are inserted into and between Steiner points in the 3D RST and construct subnets. These TSV locations are used for estimated TSV locations in the signal TSV planning flow.

5.3.1 Computation of a Die Span of a Steiner Point

The set of points of a 2D RSMT consists of fixed points (i.e., input points) and Steiner points inserted by a 2D RSMT construction algorithm. In Figure 59(c), for example, p_0 to p_5 are fixed points and s_1 to s_4 are Steiner points. When a 2D RSMT is expanded to a

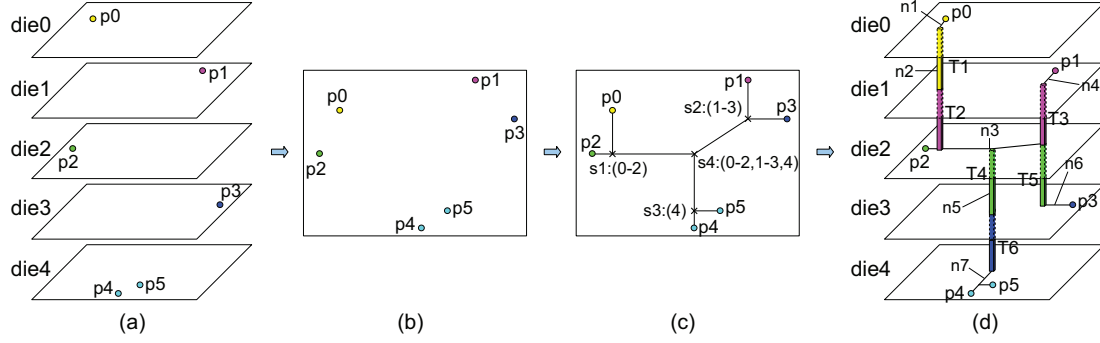


Figure 59. Construction of a 3D RST. (a) Points to be connected. (b) Fixed points projected onto a 2D xy plane. (c) A 2D RSMT. (d) A 3D RST constructed from (c).

3D RST, the 3D RST is constructed by inserting vertical edges at Steiner points as shown in Figure 59(d). However, when vertical edges are inserted into Steiner points, which dies each Steiner point connects should be determined. This problem is solved by computing a die span of each Steiner point.

To compute a die span of each Steiner point in a given 2D RSMT, a bottom-up breath-first search algorithm is applied to the 2D RSMT. In Figure 59(c), for example, depth-0 points (p0 to p5) are visited, then depth-1 points (s1, s2, s3), and then depth-2 points (s4) are visited sequentially.² The reason that the bottom-up breath-first search algorithm is applied is because the computation of die spans of higher-depth Steiner points (e.g., depth-1 points) needs determined die spans of lower-depth points (e.g., depth-0 points) adjacent to them.

Algorithm 1 shows the algorithm for the computation of a die span at each Steiner point during the 2D RSMT to 3D RST expansion. First, an empty queue, Q , is created (Line 2). Then, for each fixed point p in F , its *visited* variable is set to true (Line 4), which denotes that this point is visited and this point has a fixed die span. Its *top* and *bot* variables are also set to its die number (Line 5). For example, if a point p is located in die1 ($p.die=1$), its *top* and *bot* become 1. The *top* and *bot* variables denote the topmost die and the bottommost die that the point connects, respectively. Then these points are inserted into Q (Line 6) for

²The *depth* of a point is defined as the minimum depth from the root point set (the set of fixed points).

the breath-first search.

Between Line 8 and Line 26, the breath-first search algorithm is applied. First, a point p_1 , which is a point whose die span is already computed, is dequeued from Q (Line 9). Then, the die span of each unvisited point p_2 adjacent to p_1 is computed.³ For this, two temporary variables, $tTop$ and $tBot$, are prepared and initialized (Line 11). Then, for each visited point p_3 ⁴ adjacent to p_2 , $tTop$ is set to the smaller number of $p_3.bot$ and $tTop$ (Line 14) and $tBot$ is set to the larger number of $p_3.top$ and $tBot$ (Line 15). This computation finds the minimal die span, which connects all the visited points adjacent to p_2 , of p_2 . For example, in Figure 59(c), we first visit p_0 . Since s_1 is an unvisited point adjacent to p_0 , the die span of s_1 is computed by visiting all visited points (p_0 and p_2) adjacent to s_1 . Then, the die span of s_1 becomes $[0, 2]$ by the computation in Line 12 to Line 15 in Algorithm 1.

When the die span at Steiner point p_2 is computed, three relations between $tTop$ and $tBot$ can exist as illustrated in Figure 60. If $tTop$ is smaller than $tBot$, edge(s) connecting from $tTop$ -th die to $tBot$ -th die (Figure 60(a)) are needed. If $tTop$ equals $tBot$, no vertical edges are needed because planar edges can be used to connect visited points adjacent to s (Figure 60(b)). If $tTop$ is greater than $tBot$ as shown in Figure 60(c), there are overlaps among die spans of visited points adjacent to s , so no vertical edges are needed. In this case, a die in $[tBot, tTop]$ is chosen to connect s and visited points adjacent to s in 2D (Line 16 to Line 19). The $IRand(a, b)$ function in Line 17 returns an integer number in $[a, b]$.

Then, the die span of p_2 (Line 20) is set, and p_2 is marked as a visited point (Line 21). All unvisited points adjacent to p_2 are enqueued into Q (Line 23) for the breath-first search.

5.3.2 Insertion of TSVs into and between Steiner Points

After a 2D RSMT is expanded to a 3D RST, TSVs are inserted into and between Steiner points as follows. If top of a Steiner point is smaller than its bot ,⁵ TSVs are inserted from

³Unvisited points are always Steiner points.

⁴A visited point always has a determined die span.

⁵Notice that top is always less than or equal to bot after the die span computation.

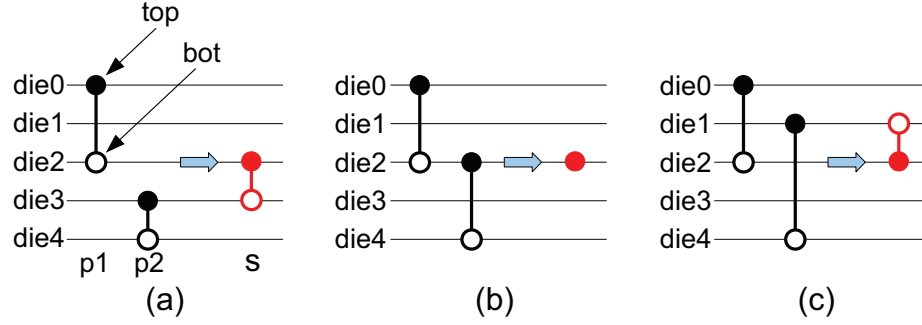


Figure 60. Die span diagrams. Solid dots are *top* variables and empty dots are *bot* variables. Red spans show $tTop$ and $tBot$ when the die span of s is computed. (a) $tTop (=2) < tBot (=3)$. (b) $tTop (=2) = tBot (=2)$. (c) $tTop (=2) > tBot (=1)$.

the (top) -th die to the $(bot-1)$ -th die.⁶ This is an insertion of TSVs into a Steiner point.

If the die spans of two adjacent Steiner points do not overlap, TSVs are also inserted between the two Steiner points. For example, if the die span of a Steiner point s_1 is $[1, 2]$ and the die span of a Steiner point s_2 adjacent to s_1 is $[4, 5]$, a TSV is inserted in die 2 and a TSV is inserted in die 3 between these two Steiner points. In this case, TSV(s) are inserted in the middle of the two points.

5.3.3 Construction of Subnets

After TSV locations for a 3D net are found, subnets are constructed for the net. For instance, the net in Figure 59(d) consists of the following subnets: n_1 connecting p_0 and the metal 1 landing pad of TSV T_1 , n_2 connecting the bottom landing pad of TSV T_1 and the metal 1 landing pad of TSV T_2 , n_3 connecting p_2 , the bottom landing pads of TSV T_2 and T_3 , and the metal 1 landing pads of TSV T_4 and T_5 , and so on.

The subnet construction algorithm is based on iterative search. For a point p in a 3D RST, an empty set S is created, p is inserted into S , and points adjacent to p are traversed. If an adjacent point j is in the same die with p , j is inserted into S . If j is in a different die, traversing through j ends. In this case, j is in the upper die, so the bottom landing pad of j is added into S . After traversing, a non-empty set S , which becomes a subnet, is found.

⁶Since it is assumed that face-to-back stacking is used, if a block in die1 is connected to another block in die3, TSVs are inserted into die1 and die2 only.

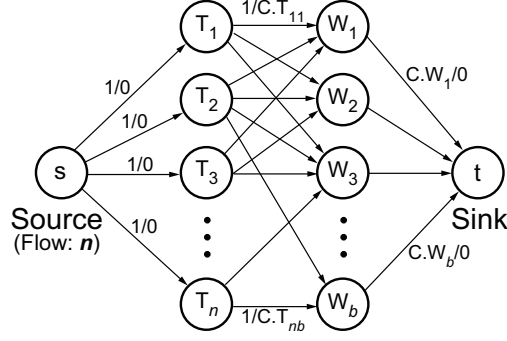


Figure 61. Global assignment of TSVs to whitespace blocks. T_i is the i -th TSV and W_j is the j -th whitespace block. f/c in each edge denotes that f is the maximum flow capacity, and c is the cost. $C.T_{ij}$ is the wirelength when TSV T_i is assigned to whitespace block W_j . $C.W_i$ is the maximum number of available TSV slots in whitespace block W_i .

This process is repeated until all the points in the 3D RST are traversed.

5.4 TSV Assignment

Since TSVs cannot be inserted into functional blocks, estimated TSV locations should be assigned to nearby whitespace blocks, as illustrated in Figure 57. To assign TSVs to whitespace blocks, a minimum-cost flow formulation is used.

5.4.1 Global TSV Assignment

Figure 61 shows the formulation for the global TSV assignment. In the figure, T_i is the node for the i -th TSV to be assigned to whitespace and W_j is the node for the j -th whitespace block. Since all TSVs should be assigned to whitespace blocks, the total amount of flow outgoing from the source equals the number of TSVs and the maximum flow capacity of each edge from the source to T_i is 1. Since edge $s \rightarrow T_i$ has no physical meaning, the cost of the edge is set to zero. Similarly, edge $W_j \rightarrow t$ has zero cost. However, the maximum flow capacity from W_j to the sink equals the number of available TSV slots in whitespace block W_j . The maximum flow capacity from T_i to W_j is 1, which denotes that a TSV is assigned to only one whitespace block. The cost of the edge $T_i \rightarrow W_j$ is computed by the Manhattan distance from T_i to W_j . This minimum-cost flow problem is formulated and solved for each die.

If the total amount of flows from whitespace blocks to the sink is less than the total number of TSVs, the problem becomes infeasible. In this case, whitespace blocks are manipulated and then the design flow goes back to the estimation of TSV locations step as illustrated in Figure 57.

5.4.2 Local TSV Assignment

After TSVs are assigned to whitespace blocks (global TSV assignment), TSVs are assigned to TSV slots in each whitespace block (local TSV assignment) in a similar way. In this local TSV assignment formulation, however, the whitespace blocks (W_j) in Figure 61 are replaced by available TSV slots (S_j) in each whitespace block and the maximum capacity of edge $S_j \rightarrow t$ is replaced by 1. The cost of edge $T_i \rightarrow S_j$ is computed by the Manhattan distance from T_i to S_j . This minimum-cost flow problem is solved for each whitespace block.

The reason that global and local assignments are applied separately is because it dramatically reduces the number of variables. If the number of variables is small, however, the TSV assignment can be performed by taking all TSVs and all TSV slots into one assignment formulation.

5.5 Whitespace Manipulation

In the signal TSV planning, whitespace manipulation is necessary in two cases. First, if assigning TSVs to whitespace blocks fails, more whitespace should be inserted. Second, even if assigning TSVs to whitespace blocks succeeds, the current floorplan could be improved further by manipulating whitespace. In this section, whitespace manipulation algorithms are presented. Although many papers use concurrent approaches [62, 63, 64], sequential whitespace manipulation (insertion, expansion, and redistribution) is adopted.

As a preparation step, whitespace is extracted for a given floorplan, four variables (*left*, *right*, *bottom*, *top*) are created for each functional block, and one variable (*demand*) is created for each whitespace block. Then, for each TSV location found, the Manhattan

Table 28. Benchmark circuits. # gates is the total number of gates in the blocks, and # nets is the total number of block-level nets.

	Circuit	# gates	# blocks	# nets	Avg. net degree
MCNC	ami33	-	33	123	4.23
	ami49	-	49	408	2.34
GSRC	n100	-	100	885	2.12
	n200	-	200	1585	2.27
	n300	-	300	1893	2.31
industrial circuits	C1	75K	51	6200	2.00
	C2	92K	98	1325	4.01
	C3	278K	46	1355	2.32
	C4	566K	47	2508	2.29

distance from the TSV to each boundary (*left, right, bottom, top*) of each functional block in the same die is computed and a demand is added to the four boundaries of the block. To compute the demand, the following function is used:

$$y = \frac{C_{\text{MAX}} - C_{\text{MIN}}}{D_{\text{MAX}} - D_{\text{MIN}}} \cdot (x - D_{\text{MIN}}) + C_{\text{MIN}} \quad (79)$$

where y is the demand, C_{MAX} is 1.0, C_{MIN} is 0.01, D_{MAX} is $W_{\text{DIE}}/6.0$, D_{MIN} to $W_{\text{DIE}}/12.0$ where W_{DIE} is the die width, and x is the distance. The Manhattan distance from each TSV location to each whitespace block in the same die is also computed and a demand is added to the *demand* variable of the whitespace block using the same demand function.

If the most demanding spot is a boundary of a functional block, a unit whitespace block, which is pre-determined by a user, is inserted to the boundary. If the most demanding spot is a whitespace block, the whitespace block is expanded by inserting a unit whitespace block to the whitespace block.

5.6 Simulation Results

The algorithms are implemented using C/C++ and perform all experiments in a 64-bit Linux server with Intel 2.5GHz CPU. To compare our algorithm with [3], MCNC and GSRC benchmarks are used. Four industry circuits are also utilized for more realistic simulation. Table 28 shows profiles of all the benchmark circuits. Since our algorithms are

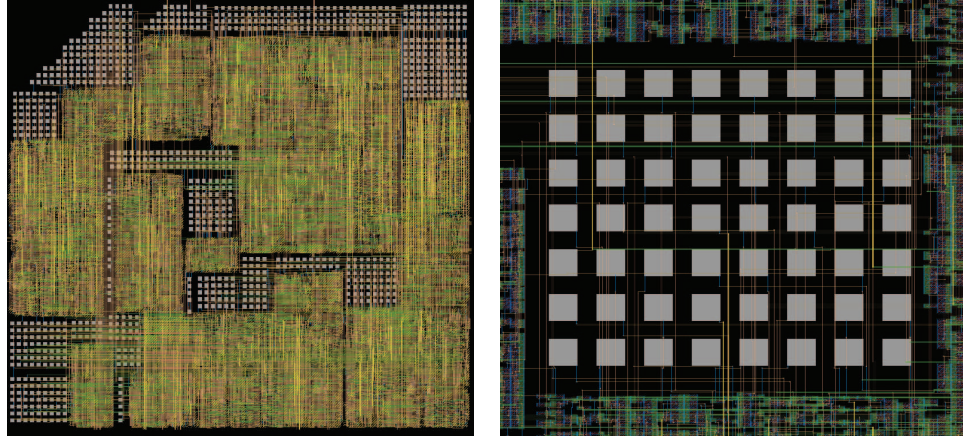


Figure 62. Full die (top-die) and zoom-in shot of four-die block-level 3D floorplanning (Cadence Virtuoso)

used in post-floorplanning steps, an in-house 3D floorplanner is developed using simulated annealing and 2D sequence pair with inter-die move as well as intra-die perturbation⁷ to generate 3D floorplans. Figure 62 shows a snapshot of the topmost die of a C2 design implemented in four dies.

5.6.1 2D Floorplanning vs 3D Floorplanning

Since all existing works on the comparison of 2D and 3D floorplans use HPWL-3DBB to estimate 3D wirelength, they do not fairly compare 2D and 3D floorplans because HPWL-3DBB significantly underestimates 3D wirelength. In addition, some of them even do not take locations of signal TSVs into account. In this experiment, therefore, HPWL of 2D floorplans and HPWL-3D of 3D floorplans post-processed by our signal TSV planner are compared. To generate 2D floorplans, our floorplanner is run in a 2D mode. To the best of our knowledge, this is the first work on the comparison of 2D and 3D floorplans using the most accurate 3D wirelength metric.

Table 29 shows that the wirelength (HPWL-3D) of 3D floorplans is slightly longer than that of 2D floorplans by 3% to 8% for relatively small circuits such as C1 and C2. However, the wirelength of 3D floorplans is much shorter than that of 2D floorplans by approximately

⁷Each die has its own sequence pair.

Table 29. Comparison of 2D and 3D floorplanning on industrial circuits. The wirelength unit is meter. Numbers in parentheses show ratios between 3D and 2D wirelengths. The TSV diameter is $2.5\mu\text{m}$, the TSV pitch is $4.0\mu\text{m}$, and the TSV length is $20.0\mu\text{m}$.

Circuit	2D	# dies	3D		
	HPWL		HPWL-3DBB	HPWL-3D	# TSVs
C1	1.515 (1.00)	2	1.042 (0.69)	1.621 (1.07)	3,080
		3	0.990 (0.65)	1.408 (0.93)	3,976
		4	0.834 (0.55)	1.595 (1.05)	5,864
		5	0.744 (0.49)	1.630 (1.08)	6,169
Geo. mean	(1.00)		(0.59)	(1.03)	
C2	0.375 (1.00)	2	0.274 (0.73)	0.366 (0.98)	1,492
		3	0.221 (0.59)	0.359 (0.96)	2,463
		4	0.198 (0.53)	0.422 (1.13)	3,837
		5	0.174 (0.47)	0.484 (1.29)	4,446
Geo. mean	(1.00)		(0.57)	(1.08)	
C3	0.819 (1.00)	2	0.522 (0.64)	1.380 (0.68)	778
		3	0.369 (0.45)	0.557 (0.68)	1,261
		4	0.404 (0.49)	0.536 (0.65)	1,337
		5	0.332 (0.40)	0.647 (0.79)	2,518
Geo. mean	(1.00)		(0.49)	(0.70)	
C4	2.094 (1.00)	2	1.423 (0.68)	1.479 (0.71)	1,226
		3	1.294 (0.62)	1.496 (0.71)	1,585
		4	1.161 (0.55)	1.491 (0.71)	2,529
		5	0.917 (0.44)	1.320 (0.63)	3,255
Geo. mean	(1.00)		(0.56)	(0.69)	

30% on average for relatively big circuits such as C3 and C4. The reason that 3D floorplans could have longer wirelength than 2D floorplans is twofold. If there are many 3D nets in a 3D floorplan, it is necessary to insert many TSVs, which could significantly increase the die area. The increased die area leads to longer inter-block connections. In addition, if inter-block connections in 2D designs are short, designing this circuit in 3D does not result in shorter inter-block connections.

One thing to notice is that HPWL-3DBB significantly underestimates 3D wirelength. In Table 29, HPWL-3DBB is 18% to 47% shorter than HPWL-3D on average. Therefore, HPWL-3D should be used as a wirelength metric for the 3D IC design.

Table 30. Comparison of signal TSV planners. Ratios between our results and [3] (Ours/[3]) are reported.

# dies	Circuit	WL	# TSVs	# dies	Circuit	WL	# TSVs
3	ami33	0.91	1.26	4	ami33	0.91	1.96
	ami49	0.78	1.26		ami49	0.70	1.34
	n100	0.93	1.03		n100	0.91	1.06
	n200	0.62	0.80		n200	0.82	1.14
	n300	0.75	0.80		n300	0.66	0.82
Geo. mean		0.79	1.01	Geo. mean		0.79	1.21

5.6.2 Comparison of Signal TSV Planners

Table 30 shows comparison of wirelength and the number of TSVs between our signal TSV planner and [3]. Since the authors of [3] use HPWL-3DBB, HPWL-3DBB is used as the wirelength metric for fair comparison. The same TSV size as [3] uses is also used. The TSV diameter for MCNC circuits is $20\mu m$ and that for GSRC circuits is $3\mu m$. Since [3] performs signal TSV insertion on fixed-outline floorplans, our 3D floorplanning is run under same constraints – fixed-outline floorplanning with the same whitespace area. I/O pin locations are also taken into the wirelength computation.

As Table 30 shows, our signal TSV planner outperforms [3] by 21% with respect to wirelength for both three-die and four-die floorplans. In addition, the difference between the wirelength of ours and that of [3] increases as the circuit size goes up. For example, ours outperforms [3] by 9% for ami33. However, for ami49, which is much bigger than ami33, the wirelength of our algorithm is 22% to 30% shorter than that of [3]. A similar trend is observed for GSRC circuits. For n100, the wirelength of ours is 7% to 9% shorter than that of [3], but for n200 or n300, ours outperforms [3] by 18% to 38%. Therefore, it is observed that our signal TSV planner optimizes wirelength more effectively than [3] as the circuit size goes up.

Since multiple TSV insertion is used, however, more TSVs are used than [3]. As Table 30 shows, 26% to 96% more TSVs are used for relatively small circuits such as ami33. However, for large circuits such as n200 and n300, slightly more TSVs or even fewer TSVs

Table 31. Comparison of single TSV insertion, 3D MST-based multiple TSV insertion, and 3D RST-based multiple TSV insertion.

	degree	Single TSV insertion		Multiple TSV insertion (3D MST-based)		Multiple TSV insertion (3D RST-based)	
		HPWL-3D ($\times 10^6$)	# TSVs	HPWL-3D ($\times 10^6$)	# TSVs	HPWL-3D ($\times 10^6$)	# TSVs
n100	3	0.209 (1.00)	1,043 (1.00)	0.168 (0.81)	1,349 (1.29)	0.156 (0.75)	1,165 (1.12)
	4	0.286 (1.00)	1,335 (1.00)	0.226 (0.79)	2,215 (1.66)	0.208 (0.73)	1,841 (1.38)
	5	0.382 (1.00)	1,415 (1.00)	0.294 (0.77)	2,779 (1.96)	0.258 (0.68)	2,258 (1.60)
	6	0.408 (1.00)	1,525 (1.00)	0.329 (0.81)	3,539 (2.32)	0.293 (0.72)	2,826 (1.85)
	7	0.472 (1.00)	1,544 (1.00)	0.439 (0.92)	4,063 (2.63)	0.356 (0.75)	3,256 (2.11)
	8	0.506 (1.00)	1,633 (1.00)	0.483 (0.95)	4,987 (3.05)	0.385 (0.76)	4,004 (2.45)
Geo. mean		(1.00)	(1.00)	(0.87)	(2.38)	(0.73)	(1.69)
n200	3	0.685 (1.00)	2,918 (1.00)	0.621 (0.91)	3,906 (1.34)	0.539 (0.79)	3,274 (1.12)
	4	0.964 (1.00)	3,544 (1.00)	0.692 (0.72)	5,800 (1.64)	0.609 (0.63)	4,771 (1.35)
	5	1.225 (1.00)	3,816 (1.00)	0.855 (0.70)	7,538 (1.98)	0.757 (0.62)	5,981 (1.57)
	6	1.385 (1.00)	4,241 (1.00)	0.949 (0.69)	9,825 (2.32)	0.832 (0.60)	7,950 (1.87)
	7	1.544 (1.00)	4,287 (1.00)	1.085 (0.70)	11,237 (2.62)	0.946 (0.61)	8,975 (2.09)
	8	1.790 (1.00)	4,516 (1.00)	1.273 (0.71)	13,742 (3.04)	1.017 (0.57)	11,127 (2.46)
Geo. mean		(1.00)	(1.00)	(0.75)	(2.39)	(0.63)	(1.65)
n300	3	1.035 (1.00)	3,703 (1.00)	0.993 (0.96)	4,876 (1.32)	0.886 (0.86)	4,111 (1.11)
	4	1.685 (1.00)	4,609 (1.00)	1.234 (0.73)	7,538 (1.64)	1.096 (0.65)	6,202 (1.35)
	5	1.671 (1.00)	4,916 (1.00)	1.172 (0.70)	9,860 (2.01)	1.027 (0.61)	7,844 (1.60)
	6	1.933 (1.00)	5,231 (1.00)	1.381 (0.71)	12,203 (2.33)	1.188 (0.61)	9,745 (1.86)
	7	2.105 (1.00)	5,430 (1.00)	1.635 (0.78)	14,449 (2.66)	1.437 (0.68)	11,536 (2.12)
	8	2.362 (1.00)	5,543 (1.00)	2.132 (0.90)	16,865 (3.04)	1.633 (0.69)	13,394 (2.42)
Geo. mean		(1.00)	(1.00)	(0.79)	(2.38)	(0.68)	(1.68)

are used. Since 3D floorplanning has a great effect on the number of TSVs used by signal TSV planners, this result also shows that our 3D floorplanner outperforms the 3D floorplanner used in [3].

5.6.3 Single TSV Insertion vs. Multiple TSV Insertion

Multiple TSV insertion can reduce wirelength further than single TSV insertion. In this experiment, therefore, single TSV insertion, 3D minimum spanning tree (MST)-based multiple TSV insertion, and 3D RST-based multiple TSV insertion are compared. For the single TSV insertion, a single TSV insertion algorithm similar to [3] is used. For the multiple TSV insertion, since the 3D MST is frequently used to find TSV locations [11], a multiple TSV insertion algorithm using the 3D MST is implemented. In this algorithm, a 3D MST is created for each 3D net, and then each 3D edge is converted into TSV(s), similarly as shown in [11]. In addition, since multiple TSV insertion improves total wirelength effectively for high-degree nets, benchmarks having n nets of degree d are generated. In Table 31, for example, n100 with average net degree 5 denotes that it has 576 nets, and

each net is of degree 5.

Table 31 shows wirelength and the number of TSVs of these three signal TSV insertion algorithms. As the table shows, 3D MST-based multiple TSV insertion leads to 13% to 25% shorter wirelength on average than the single TSV insertion. In addition, 3D RST-based multiple TSV insertion produces 27% to 37% shorter wirelength on average than the single TSV insertion.

However, since multiple TSV insertion inserts more TSVs than single TSV insertion, the 3D MST-based multiple TSV insertion inserts $2.38\times$ more TSVs on average than the single TSV insertion. Similarly, the 3D RST-based multiple TSV insertion inserts $1.67\times$ more TSVs on average than the single TSV insertion. However, the 3D RST-based multiple TSV insertion uses much less number of TSVs (30% on average) than the 3D MST-based multiple TSV insertion. Therefore, using 3D RST to find optimal TSV locations results in less TSVs and shorter wirelength than using 3D MST.

In Table 31, it is also observed that wirelength reduction increases as the average net degree goes up. If all nets are two-pin nets (degree 2), no difference exists between single TSV insertion and multiple TSV insertion. However, if all nets are high-degree multi-pin nets (e.g., degree 5), using multiple TSVs helps reduce the total wirelength.

5.7 Summary

In this chapter, a signal TSV planning method is proposed to effectively insert signal TSVs into whitespace for the design of block-level 3D ICs. The signal TSV planning flow estimates TSV locations, assigns the estimated TSV locations to existing whitespace blocks, and manipulates (insertion, expansion, and redistribution) whitespace blocks. Estimation of TSV locations uses 3D rectilinear Steiner tree to find potential TSV locations minimizing 3D wirelength. The simulation results show that the proposed signal TSV planning method outperforms other signal TSV planning methods. In addition, the 3D RST-based

multiple TSV insertion algorithm outperforms the single TSV insertion and the 3D MST-based multiple TSV insertion in terms of wirelength and the number of TSVs inserted.

CHAPTER 6

THE IMPACT OF THROUGH-SILICON VIAS ON 3D INTEGRATED CIRCUITS

Three-dimensional integrated circuits (3D ICs) are expected to offer various benefits such as higher bandwidth, smaller form factor, shorter wirelength, lower power, and better performance than traditional two-dimensional (2D) ICs. These benefits are enabled by die stacking and the use of through-silicon vias (TSVs) for inter-die connections. However, TSVs have two negative effects, occupation of silicon area and non-negligible capacitance, in the design of 3D ICs. The fact that TSVs occupy silicon area has great effects not only on silicon area, but also on wirelength, critical path delay, and power. The reason is as follows. If larger TSVs are inserted in a 3D IC layout, footprint area of the design becomes larger, so the average wirelength increases [65]. This wirelength overhead leads to longer critical path delay and higher dynamic power consumption due to increased wire capacitance. In addition, non-negligible TSV capacitance also has a negative effect on critical path delay and dynamic power consumption. One thing to notice is that smaller TSVs do not necessarily have smaller capacitance than larger TSVs. The reason is because TSV capacitance is dependent not only on the TSV diameter and the TSV height, but also on the liner thickness and doping concentration of the substrate [30].

Similarly as devices are scaled, TSVs are also being downscaled [66, 67, 68]. Therefore, negative effects of TSVs will be reduced if smaller TSVs are used.¹ However, this statement is valid only when process technology is fixed and TSV technology advances. In fact, process technology is also advancing, so it is highly likely that future 3D ICs will be fabricated with smaller TSVs and state-of-the-art process technology. In this case, negative effects of TSVs might remain the same or even increase.

¹If it is assumed that only the TSV size and the TSV height are downscaled while other design parameters such as the liner thickness and doping concentration are fixed, TSV capacitance decreases as TSVs are downscaled.

In this research, the impact of sub-micron TSVs on area, wirelength, critical path delay, and power of today and future 3D ICs is investigated based on GDSII-level layouts. For future process technologies, $22nm$ and $16nm$ process and standard cell libraries are developed. With these future process technologies as well as an existing $45nm$ library, 3D IC layouts are generated with different TSV sizes and capacitances and study the impact of TSVs thoroughly. The following contributions are made in this research:

- To investigate the impact of sub-micron TSVs on future 3D ICs, a $22nm$ and a $16nm$ process and standard cell libraries are developed. These libraries enable us to obtain very trustable experimental results.
- Layouts with various technology combinations (e.g., $16nm$ process with $0.5\mu m$ -diameter TSVs and $0.1\mu m$ -diameter TSVs) are generated and area, wirelength, critical path delay, and power of the layouts are obtained. Therefore, 3D ICs built with different process technologies are cross-compared and 3D ICs built with a same process technology and different TSV sizes and capacitances are compared too.
- 2D designs built with more advanced process technology and 3D designs built with older process technology are compared. Simulation results show that 3D ICs built with an n -th generation process technology could be beaten by 2D ICs built with an $n + 2$ -th generation process technology.²

6.1 Preliminaries

6.1.1 Negative Effects of TSVs

The use of TSVs in 3D ICs have two negative effects on the quality of 3D ICs: area and delay overhead. According to recent research on TSV area overhead [33], silicon area occupied by TSVs is quite significant, which in turn reduces the wirelength benefit of 3D ICs. In addition, according to recent research on TSV capacitance overhead [47], TSV capacitance is a significant source of delay on 3D signal paths. Although buffer insertion

²This observation is strongly dependent on TSV capacitance used at each process node.

can reduce delay overhead caused by TSV capacitance, buffer insertion itself also causes another problem: additional silicon area for buffer insertion and additional power consumption.

The degree of negative effects of TSVs on 3D ICs is dependent on various technology and design parameters. For example, if $5\mu\text{m}$ TSVs³ are used with state-of-the-art process technology such as 32nm technology in 3D IC designs, these TSVs may cause a huge area overhead. On the other hand, if $5\mu\text{m}$ TSVs are used with relatively old technology such as $0.18\mu\text{m}$ technology, these TSVs may not cause any area overhead because the ratio between area occupied by a TSV and average gate area of the old technology is smaller than the ratio of the advanced technology. Similarly, small TSVs (e.g., $1\mu\text{m}$ TSVs) can have huge capacitance depending on the liner thickness and doping concentration of the substrate. In this case, small TSVs may not cause serious area overhead, but they will cause serious delay overhead.

6.1.2 Motivation

Downscaling of devices reached 32nm node [6] in 2009, and 22nm and 16nm technologies are currently under development. As devices are downscaled as process technology advances, TSVs are also being downscaled as TSV manufacturing technology advances. Recently, it was demonstrated that $0.7\mu\text{m}$ -diameter TSVs could also be fabricated reliably [68]. In addition, according to the ITRS prediction on TSV diameter and TSV aspect ratio, TSV diameter will continue to decrease while TSV aspect ratio will increase. Therefore, it is expected that sub-micron TSVs will be developed and be ready for use within the next few years.

However, all of the existing work on the impact of TSVs on the quality of 3D IC designs focus on using micron-size TSVs and current (32nm or 45nm) or even old (90nm and 130nm) process technology. For example, a 45nm technology and $1.67\mu\text{m}$ TSVs are

³A “ $X\mu\text{m}$ TSV” denotes a TSV whose width (= for square-shaped TSVs) or diameter (= for cylindrical-type TSVs) is $X\mu\text{m}$.

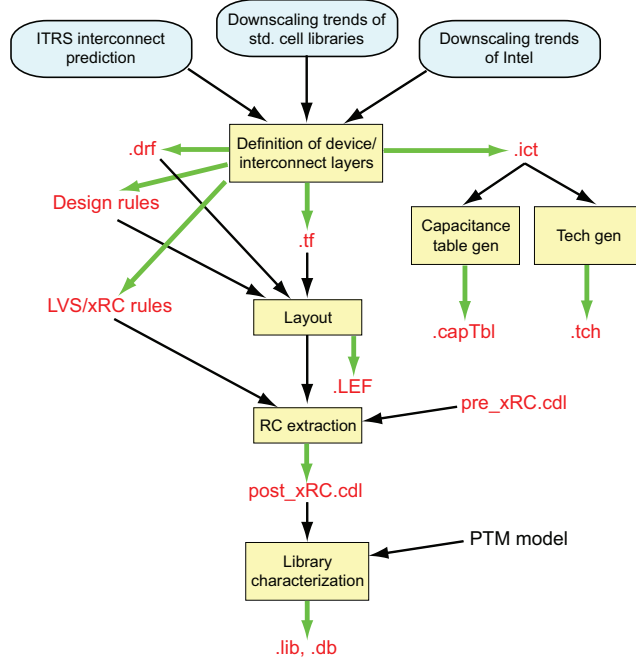


Figure 63. Development flow of our 22nm and 16nm process and standard cell libraries.

used in [11] and a 45nm technology and TSVs whose width is approximately $4\mu\text{m}$ are used in [58]. However, none of them discuss what will occur if smaller TSVs are used in 45nm technology or what will occur if the same-size TSVs are used with different process technologies (e.g., a 90nm, 32nm, or 22nm). However, it is crucial to accurately predict the impact of new TSV technology on the design quality of 3D ICs in order to refine the technology or justify the investment and cost. Our goal in this paper is to study the impact of sub-micron TSVs on the area, wirelength, critical path delay, and power of today and future 3D IC designs. For our future process technology, a 22nm and a 16nm process and standard cell libraries are developed. Various sets of TSV-related dimensions are also used in the GDSII-level 3D IC layouts. Lastly, a thorough study on the impact of sub-micron TSVs on the design quality of today and future 3D ICs is presented.

6.2 Library Development Flow

In this section, the development flow of our 22nm and 16nm process and standard cell libraries is demonstrated. For 22nm and 16nm transistor models, the high-performance

Table 32. Interconnect layers of 65nm [4], 45nm [5], 32nm [6], 22nm, and 16nm process technology. The 22nm and the 16nm layers are from our prediction.

Layer	Pitch (nm)				
	65nm	45nm	32nm	22nm	16nm
Contacted Gate	220	160	112.5	86	62
Metal 1	210	160	112.5	76	46
Metal 2	210	160	112.5	76	46
Metal 3	220	160	112.5	76	46
Metal 4	280	240	168.8	130	72
Metal 5	330	280	225.0	206	98
Metal 6	480	360	337.6	206	146
Metal 7	720	560	450.1	390	240
Metal 8	1080	810	566.5	390	240

Table 33. Width (w) and thickness (t) of metal layers used in our 22nm and 16nm process libraries. The aspect ratio for the 22nm library is 1.8 and that for the 16nm library is 1.9.

Layer	22nm		16nm	
	w (nm)	t (nm)	w (nm)	t (nm)
Metal 1, 2, 3	36	64.8	22	41.8
Metal 4	60	108	32	60.8
Metal 5	96	172.8	44	83.6
Metal 6	96	172.8	66	125.4
Metal 7, 8	180	324	110	209
Metal 9, 10	400	720	400	760
Metal 11, 12	800	1440	800	1520

transistor model of the predictive technology model (22nm PTM HP model V2.1 and 16nm PTM HP model V2.1) is used [69]. The supply voltages of the 22nm and the 16nm models are 0.8V and 0.7V, respectively.

6.2.1 Overall Development Flow

For the development of a 22nm and a 16nm process and standard cell libraries, a typical library development flow illustrated in Figure 63 is used. First, device and interconnect layers are defined. From the defined device and interconnect layers, a tech file (.tf), a display resource file (.drf), an interconnect technology file (.ict), a design rule file, a layout-versus-schematic (LVS) rule file, and an RC parasitic extraction rule file are created. With

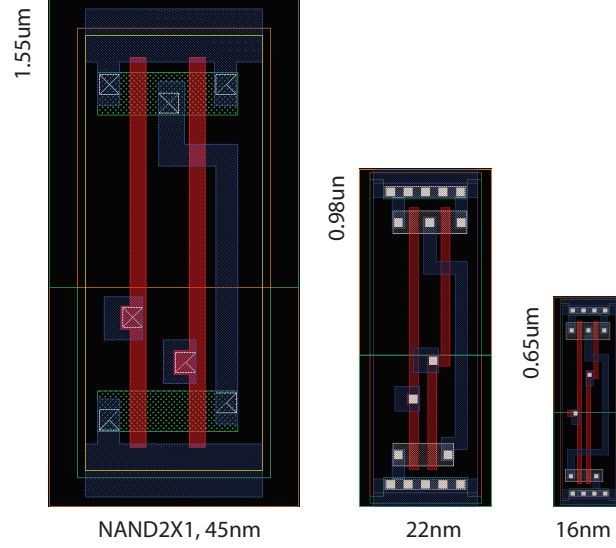


Figure 64. The smallest ($1\times$) two-input NAND gates of the 45nm [12], and our 22nm and 16nm libraries (drawn to scale).

the tech file and the display resource file, standard cell layouts are drawn. After the layout generation, abstraction is performed on these layouts to create a library exchange format file (.LEF), and run RC extraction and create SPICE netlists (post_xRC.cdl). With these SPICE netlists and the PTM transistor models, library characterization is performed to create timing and power libraries (.lib and .db). A capacitance table and a .tch file are also generated for sign-off RC extraction and timing analysis.

6.2.2 Interconnect Layers

Interconnect layers of the 22nm and 16nm technology are created based on ITRS interconnect prediction [70], downscaling trends of other standard cell libraries, and the downscaling trends of Intel process technology [4, 5, 6]. According to ITRS prediction on interconnect layers, for example, the pitch of the metal 1 wire at 22nm is about 72nm and that at 16nm is about 48nm, and the pitch of a semi-global wire at 22nm is about 160nm and that at 16nm is about 130nm. From these values as well as extrapolation of interconnect layers of Intel process technology and other standard cell libraries, interconnect layers at 22nm and 16nm are predicted. Table 32 shows the contacted gate pitch and the pitches of metal 1 to metal 8 layers at each process node. Table 33 shows widths and thicknesses

of all metal layers of our $22nm$ and $16nm$ process libraries. Notice that the $22nm$ and the $16nm$ libraries have the same width in metal 9 to metal 12 layers. Since these metal layers are sometimes used for special purposes such as power/ground lines and clock distribution, they are not scaled down. The aspect ratio of the $22nm$ library is set to 1.8 and that of the $16nm$ library is set to 1.9. Since it is assumed that the use of low-k inter-layer insulator material, 1.9 is used for the dielectric constant of the inter-layer dielectric material and 3.8 for the dielectric constant of the barrier material for both the $22nm$ and the $16nm$ libraries.

6.2.3 Standard Cell Library

First, a tech file defining device and interconnect layers and a set of design rules such as minimum poly-to-contact spacing, minimum metal-to-metal spacing, and so on, are created. Then, standard cell layouts are drawn with this tech file and the design rules. About 90 cells are created and Table 34 lists the standard cells except antenna and filler cells. The placement site width and height of our $22nm$ standard cell library are $0.1\mu m$ and $0.9\mu m$, respectively, and those of our $16nm$ library are $0.06\mu m$ and $0.6\mu m$, respectively. Figure 64 shows the smallest ($1\times$) two-input NAND gates of the $45nm$, our $22nm$, and $16nm$ standard cell libraries. After creating the standard cell layouts, DRC and LVS are performed for each layout and parasitic RC of each standard cell is extracted. All the standard cells are also characterized to create timing and power libraries.

6.2.4 Comparison of $45nm$, $22nm$, and $16nm$ Libraries

In this section, the Nangate $45nm$, our $22nm$, and our $16nm$ standard cell libraries and transistor characteristics are compared.

6.2.4.1 Gate Delay and Input Capacitance

Gate delay and drive strength are determined by transistor characteristics and the gate size. Therefore, our first experiment is to compare the transistor characteristics. In this experiment, a minimum-size inverter in a process library drives another minimum-size inverter, which drives an $N\times$ inverter in the same library. The delay of the second minimum-size

Table 34. Standard cells in our 22nm and 16nm standard cell libraries.

Type	Available sizes
AND2/3/4, AOI21/211/221	1×, 2×, 4×
BUF, INV	1×, 2×, 4×, 8×, 16×, 32×
LOGIC 0, LOGIC 1	1×
MUX2	1×, 2×
NAND2/3/4/, NOR2/3/4	1×, 2×, 4×
OAI21/22/211/221/222	1×, 2×, 4×
OAI33	1×
OR2/3/4	1×, 2×, 4×
XNOR2, XOR2	1×, 2×
DFF	1×, 2×
FA, HA	1×

Table 35. FO4 delay, standard cell heights, wire sheet resistance, and unit wire capacitance ($fF/\mu m$).

	45nm	22nm	16nm
FO4 delay	15.15ps	13.63ps	12.28ps
Std. cell. height	1.4 μm	0.9 μm	0.6 μm
Wire sheet resistance (Metal 1)	0.38	0.26	0.40
(Metal 4)	0.21	0.16	0.28
(Metal 7)	0.08	0.05	0.08
Unit wire capacitance (Metal 1)	0.20	0.15	0.16
(Metal 4)	0.20	0.15	0.13
(Metal 7)	0.20	0.14	0.14

inverter (driving the $N\times$ inverter) is obtained by SPICE simulation. Figure 65 shows the delay. It is observed that the 16nm inverter has the shortest delay and the 45nm inverter has the longest delay. Quantitatively, approximately 30% improvement is observed when the process moves from 45nm to 22nm and about 20% improvement is observed when the process moves from 22nm to 16nm. Notice that this SPICE simulation does not consider interconnect parasitic resistance and capacitance. Table 35 also shows the FO4 delay at each process technology.

Since gate input capacitance is also an important factor determining delay and power, input capacitances of 45nm, 22nm, and 16nm standard cells are presented in Table 36. As

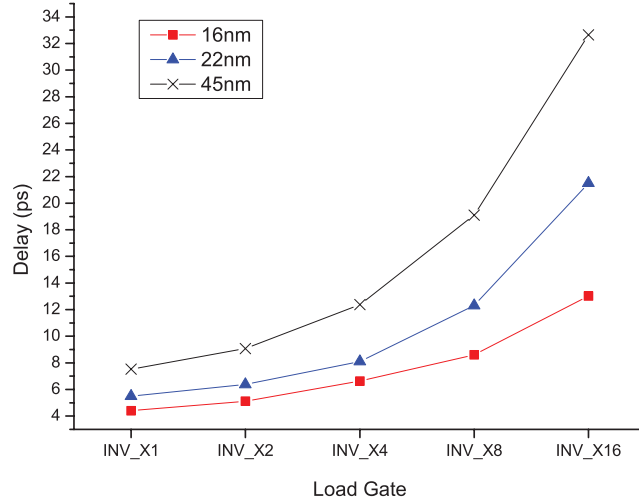


Figure 65. Delay of a minimum-size inverter driving an $N \times$ inverter ($N = 1, 2, 4, 8, 16$), where both inverters are in the same process. RC parasitics are included.

shown in the table, the average input capacitance of the 22nm standard cells is approximately 48% of the average input capacitance of the 45nm standard cells. On the other hand, the average input capacitance of the 16nm standard cells is approximately 83% of the average input capacitance of the 22nm standard cells.

6.2.4.2 Interconnect Layers

Characteristics of interconnect layers also have a big effect on the performance of a library, so wire sheet resistance and unit wire capacitance of short, semi-global, and global metal layers are listed in Table 35. The resistivity of the 45nm technology is about 5.0×10^{-8} , so the sheet resistance of the library is relatively high compared to the 22nm library. On the other hand, the resistivity of the 22nm and the 45nm technology is 1.7×10^{-8} , which is the resistivity of copper. This is why the sheet resistances of the 22nm metal layers are lower than those of the 45nm metal layers although the thickness of the 45nm metal layers is larger than that of the 22nm metal layers. On the other hand, as the technology moves from 22nm to 16nm, the sheet resistance goes up because both of them use the same resistivity, but the metal layer thickness of the 16nm library is smaller than that of the 22nm library.

The unit wire capacitance of the 45nm library is also slightly higher than that of the 22nm library. This is because the dielectric constant used for the 45nm library is 2.5 while

Table 36. Input capacitance of selected standard cells in the 45nm, the 22nm, and the 16nm libraries.

Cell	Cap (fF)		
	45nm	22nm	16nm
AND2 1×	0.54 (1.00)	0.25 (0.46)	0.22 (0.41)
AOI211 1×	0.64 (1.00)	0.30 (0.47)	0.25 (0.39)
AOI21 1×	0.55 (1.00)	0.23 (0.42)	0.20 (0.36)
BUF 4×	0.47 (1.00)	0.28 (0.60)	0.29 (0.62)
DFF 1×	0.90 (1.00)	0.41 (0.46)	0.26 (0.29)
FA 1×	2.46 (1.00)	1.31 (0.53)	1.36 (0.55)
INV 4×	1.45 (1.00)	0.69 (0.48)	0.56 (0.39)
MUX2 1×	0.95 (1.00)	0.42 (0.44)	0.34 (0.36)
NAND2 1×	0.50 (1.00)	0.24 (0.48)	0.22 (0.44)
OAI21 1×	0.53 (1.00)	0.25 (0.47)	0.20 (0.38)
OR2 1×	0.60 (1.00)	0.26 (0.43)	0.20 (0.33)
XOR2 1×	1.08 (1.00)	0.55 (0.51)	0.45 (0.42)
Average	(1.00)	(0.48)	(0.40)

Table 37. Benchmark circuits.

Circuit	# Gates	# Nets	Total cell area		
			45nm	22nm	16nm
BM1	352K	372K	0.632	0.218	0.098
BM2	518K	680K	1.288	0.437	0.198

the 22nm library uses 1.9 for its dielectric constant. If the same dielectric material ($\epsilon_r=1.9$) is used for the 45nm library, the unit wire capacitance becomes 0.15, which is close to the unit wire capacitance of the 22nm library.

6.2.4.3 Full-Chip 2D Design

In this experiment, 2D circuits are designed using the three standard cell libraries and compare area, wirelength, critical path delay, and power. The experimental flow is as follows. Two benchmark circuits shown in Table 37 are synthesized, designed, and optimized using each standard cell library and commercial tools. For all the libraries, the same area utilization (60%) is used for fair comparison and the fastest operation frequency is found for each library.

Table 38 shows the comparison results for the 2D designs. The chip area of the 45nm designs is about three times larger than that of the 22nm designs on average, and the chip

Table 38. Comparison of 2D layouts.

	BM1			BM2		
	45nm	22nm	16nm	45nm	22nm	16nm
Area (mm^2)	1.00	0.36	0.17	2.56	0.81	0.42
Wirelength (m)	10.65	4.22	2.75	15.17	8.90	6.19
Delay (ns)	3.19	2.61	2.38	6.51	4.10	3.93
Power (W)	0.352	0.0684	0.068	0.521	0.154	0.133

area of the 22nm designs is approximately two times larger than that of the 16nm designs on average. In addition, the total wirelength of the 16nm designs is approximately 1.48× shorter than that of the 22nm designs, and 3.08× shorter than that of the 45nm designs. Regarding the critical path delay, the 16nm designs are 1.49× faster than the 45nm designs on average and 1.07× faster than the 22nm designs on average. Power consumption of the 16nm designs is approximately 4.5× smaller than that of the 45nm designs and 1.1× smaller than that of the 22nm designs. Overall, the delay and power enhancement coming from 22nm-to-16nm transition is not as significant as the enhancement coming from 45nm-to-22nm transition because 45nm and 22nm technologies are two generations apart while 22nm and 16nm technologies are only one generation apart, and the quality (sheet resistance and unit wire capacitance) of the interconnect layers of the 45nm library is worse than that of the 22nm library.

6.3 Full-Chip 3D IC Design and Analysis Methodology

To generate 3D IC layouts, the 3D RTL-to-GDSII tool obtained from [58] is used. This tool works as follows: For a given 2D gate-level (flattened) netlist, this tool partitions gates in x-, y-, and z- directions iteratively to globally place gates in grids in 3D. After the global placement, it constructs a 3D Steiner tree for each net and inserts TSVs into each placement grid based on the locations of vertical edges in the 3D Steiner tree. Then, it runs detailed placement in each placement grid using Cadence Encounter. Routing for each die is also performed by Encounter. The output of the tool includes a verilog netlist

and a design exchange format (DEF) file containing TSV locations for each die, a top-level verilog netlist containing die-to-die connections, and a top-level standard parasitic exchange format (SPEF) file. One thing to notice is that the minimum number of TSVs to be inserted in the 3D design is dependent on the cut sequence, which is the order of x-, y-, and z- direction partitioning applied during global placement. For example, if z-direction partitioning is applied in early steps, it is likely that fewer inter-die connections are obtained. On the other hand, if z-direction partitioning is applied later, it is likely that more inter-die connections are obtained [58]. This variation of the number of TSVs enables us to produce different global placement solutions with different TSV counts.

After 3D IC layout generation, 3D timing optimization is performed. First, initial timing optimization is performed for each die. Then, all the layouts, timing analysis results, and the target clock frequency are fed into the 3D timing optimization tool obtained from [71]. This 3D timing optimization tool iterates the following steps: (a) it performs RC extraction and obtains an SPEF file for each die; (b) it performs 3D timing analysis using the SPEF files and the top-level SPEF file using Synopsys PrimeTime; (c) based on the timing analysis result and the target clock frequency, the tool scales the target delay of each 3D path and creates a timing constraint file for each die; (d) since each die has its own netlist and timing constraint file, timing optimization is performed for each die separately. This timing optimization process is repeated several times until the overall timing improvement saturates.

3D power analysis needs (a) a top-level netlist as well as a netlist for each die, (b) a top-level SPEF file as well as a SPEF file for each die, and (c) switching activities of cells and nets. To obtain switching activities of cells and nets, verilog netlists generated by the 3D RTL-to-GDSII tool obtained from [58] are fed into Encounter and run power analysis. This power analysis internally generates and stores switching activities of cells and nets, so this information is dumped into an output file after the power analysis. Then, all the netlists, SPEF files, and the switching activity files are loaded into PrimeTime and run

Table 39. TSV-related dimensions, design rules, and TSV capacitance.

Dimensions	TSV-5	TSV-1	TSV-0.5	TSV-0.1
Width (μm)	5	1	0.5	0.1
Height (μm)	25	5	8	5
Aspect ratio	5	5	16	50
Liner thickness (nm)	100	30	20	10
Barrier thickness (nm)	50	15	10	5
Landing pad width (μm)	6	1.6	1	0.18
TSV-to-TSV spacing (μm)	2	0.8	0.6	0.1
TSV-to-device spacing (μm)	1	0.4	0.3	0.1
TSV capacitance (fF)	20	2.67	3.2	0.8

power analysis. This power analysis method produces true full-chip 3D power analysis results.

6.4 Simulation Results

6.4.1 Simulation Settings

Two benchmark circuits, BM1 and BM2, as shown in Table 37 are used. For the 45nm process node, the Nangate 45nm standard cell library [12] is used. Four sets of TSV-related dimensions listed in Table 39 are also used. In the simulation, 5 μm and 0.5 μm TSVs are used with the 45nm technology, 1 μm and 0.1 μm TSVs are used with the 22nm technology, and 0.5 μm and 0.1 μm TSVs are used with the 16nm technology. Since the standard cell height of the 45nm library is 1.4 μm , a 5 μm TSV including its keep-out zone occupies five standard cell rows while a 0.5 μm TSV including its keep-out zone occupies one standard cell row, as shown in Table 39. Similarly, a 1 μm TSV and a 0.1 μm TSV occupy three standard cell rows and 0.26 standard row, respectively, when they are used with our 22nm standard cell library. If 0.5 μm and 0.1 μm TSVs are used for our 16nm standard cell library, a 0.5 μm TSV occupies 1.33 standard cell rows and a 0.1 μm TSV occupies 0.5 standard cell row. Figure 66 shows the four different TSVs in a top-down view and a side view and Figure 67 shows GDSII images of TSVs and standard cells at 45nm, 22nm, and 16nm technology.

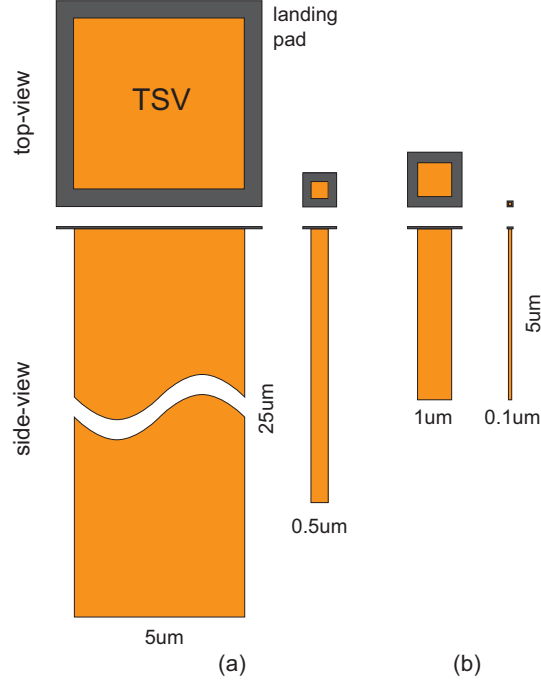


Figure 66. Size comparison of the 4 TSVs used in our study: (a) $5\mu\text{m}$ and $0.5\mu\text{m}$ width used for 45nm technology, (b) $1\mu\text{m}$ and $0.1\mu\text{m}$ width used for 22nm technology.

6.4.2 Impact on Silicon Area

Figure 68 shows footprint area of 2D designs and two-die 3D BM1 designs at each technology node. If the TSV size is zero, the footprint area of a two-die 3D design should be approximately half of its 2D counterpart. Since the TSV is not zero, however, the footprint area of a two-die 3D design is usually greater than half of its 2D counterpart. For example, the area of the 45nm 2D design is 1.0mm^2 , but the area of the 45nm 3D design using $5\mu\text{m}$ TSVs is about 0.85mm^2 , which is 85% of the 2D design. Similarly, the area of the 45nm 3D design using $0.5\mu\text{m}$ TSVs is about 0.63mm^2 , which is 63% of the 2D design. The same trend is found in the 22nm and the 16nm designs. However, if the TSV size is $0.1\mu\text{m}$, the footprint area of a two-die 3D design becomes almost half of its 2D counterpart. Similar trends are found in BM2 designs as shown in Figure 69.

All these trends depend on TSV size and the number of TSVs used in the designs. Of course, using smaller TSVs helps achieve smaller footprint area, which can reduce chip cost. However, smaller TSVs could be more expensive due to manufacturing difficulties,

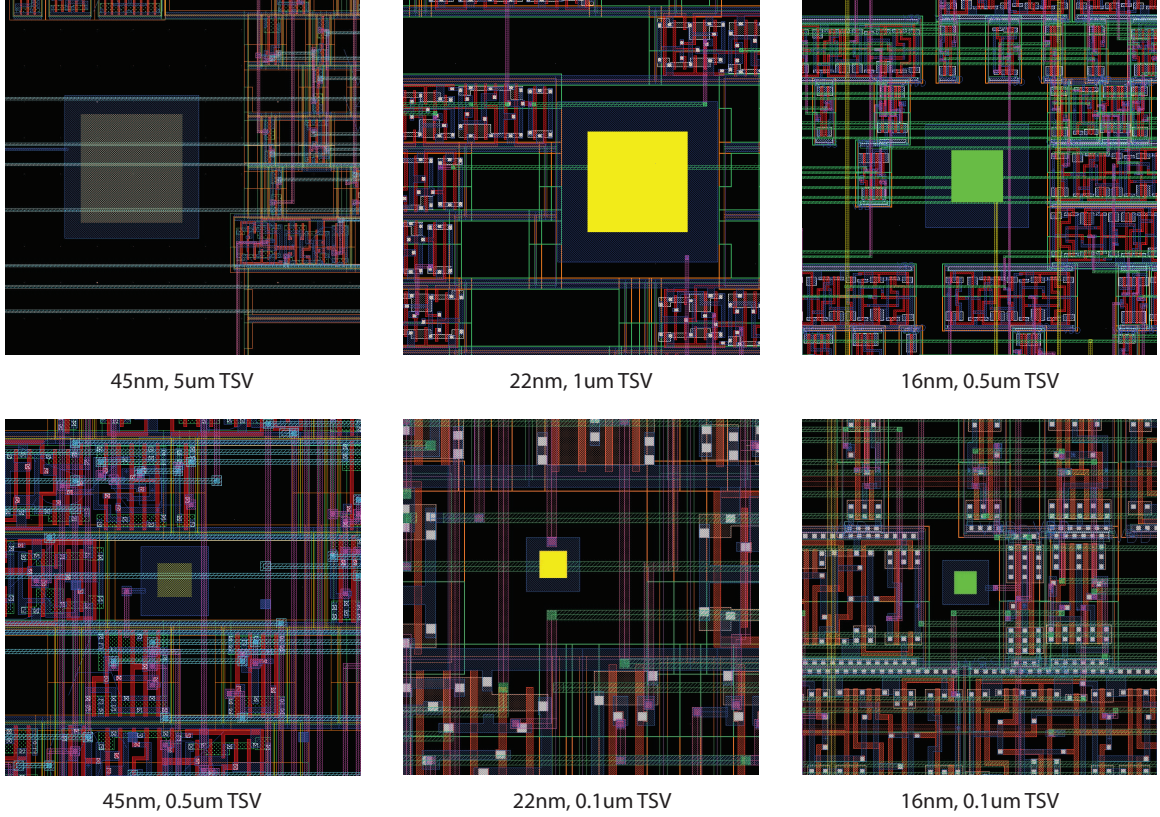


Figure 67. Zoom-in GDSII layouts of the six types of designs studied in this paper. Each TSV is surrounded by its keep-out-zone.

so the use of smaller TSVs does not necessarily leads to lower chip cost. Using fewer TSVs also helps achieve smaller footprint area. However, several studies show that using more TSVs than the minimum number of TSVs helps reduce wirelength and achieve better performance [32, 11, 58]. Thus, there exist trade-offs among TSV size, the number of TSVs used in the design, and chip cost.

6.4.3 Impact on Wirelength

Figure 68 shows wirelength of BM1 designs. When $5\mu m$ TSVs are used with the $45nm$ technology, 3D designs have longer wirelength than 2D designs. However, when $0.5\mu m$ TSVs are used with $45nm$ technology, the wirelength of the 3D design is about 10% shorter than that of the 2D design. When $1\mu m$ and $0.1\mu m$ TSVs are used with the $22nm$ technology, however, large wirelength reduction is not observed. On the other hand, when $0.5\mu m$ and $0.1\mu m$ TSVs are used with the $16nm$ technology, 15% wirelength reduction is observed.

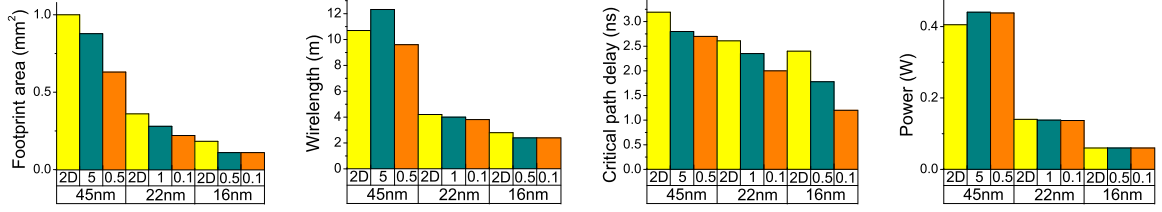


Figure 68. Comparison of the optimized 2D designs and two-die 3D designs (BM1) in 45nm, 22nm, and 16nm technology. The x-axis shows the technology combination (the first row shows TSV diameter in μm).

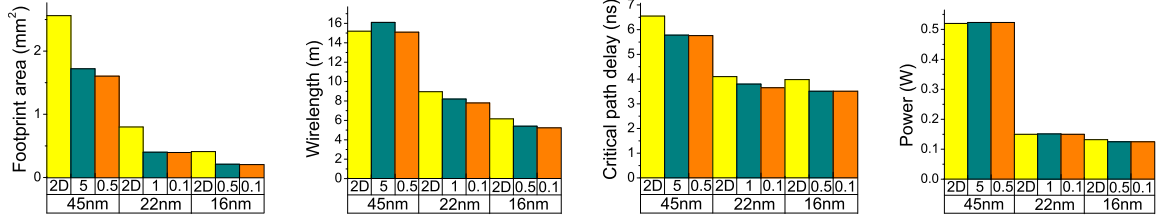


Figure 69. Comparison of the optimized 2D designs and two-die 3D designs (BM2) in 45nm, 22nm, and 16nm technology. The x-axis shows the technology combination (the first row shows TSV diameter in μm).

Similar trends are found in BM2 designs too as shown in Figure 69. Above all, 45nm 3D designs have longer wirelength than 2D designs. However, when $1\mu\text{m}$ and $0.1\mu\text{m}$ TSVs are used with the 22nm technology, 9% and 13% wirelength reduction are observed, respectively. Similarly, when $0.5\mu\text{m}$ and $0.1\mu\text{m}$ TSVs are used with the 16nm technology, 12% and 15% wirelength reduction are observed, respectively.

Wirelength reduction obtained by moving from 2D ICs to 3D ICs comes mainly from smaller footprint area. However, wirelength reduction is also dependent on the quality of the 3D global placement algorithm, the TSV insertion (3D routing) algorithm, TSV size, and characteristics of benchmarks circuits. Therefore, it is possible to obtain higher wirelength reduction ratio or vice versa depending on those factors. If, however, all other factors such as the placement algorithm and the TSV insertion algorithm except the TSV size do not change, the TSV size is the main factor affecting the wirelength. For instance, in Figure 68, by shrinking the TSV size from $5\mu\text{m}$ to $0.5\mu\text{m}$ in the 45nm 3D designs, 22% wirelength reduction is obtained. However, when the TSV size shrinks from $0.5\mu\text{m}$ to $0.1\mu\text{m}$ in the 16nm 3D designs, almost no wirelength reduction is obtained. This is because

Table 40. Additional TSV-related statistics. “c.p.” denotes critical path.

	BM1					
	45nm		22nm		16nm	
TSV diameter	5 μm	0.5 μm	1 μm	0.1 μm	0.5 μm	0.1 μm
# TSVs in c.p.	1	0	3	4	2	4

	BM2					
	45nm		22nm		16nm	
TSV diameter	5 μm	0.5 μm	1 μm	0.1 μm	0.5 μm	0.1 μm
# TSVs in c.p.	0	0	1	2	1	2

0.5 μm TSVs are already sufficiently small, so shrinking the TSV size does not lead to further wirelength reduction.

One thing to note is that 3D designs at the n -th generation process node has longer wirelength than 2D designs at the $n + 1$ -th generation process node. Therefore, shrinking the TSV size is important to reduce the wirelength, but moving to the advanced process node is also important for wirelength reduction. This also coincides with the prediction result presented in [72].

6.4.4 Impact on Performance

Figure 68 shows the critical path delay of 2D and 3D designs for the BM1 benchmark circuit. As seen in the figure, the critical path delay of a 3D design having longer wirelength than (or similar wirelength to) its 2D counterpart can be smaller than that of the 2D design. For example, the wirelength of the 3D design built with 5 μm TSVs and the 45nm technology is 15% longer than that of the 2D design, but the critical path delay of the 3D design is 12% smaller than that of the 2D design. Similar trends are also found in the BM2 benchmark circuit as shown in Figure 69.

One important observation is that the critical path delay of 3D designs built with the n -th generation process node could be smaller than the critical path delay of 2D designs built with the $n + 1$ -th generation process node. For example, the BM1 3D design built with 0.1 μm TSVs with the 22nm technology has approximately 20% smaller delay than the 2D design built with the 16nm technology. Similarly, the BM2 3D design built with 0.1 μm

TSVs with the 22nm technology has about 9% smaller delay than the 2D design built with the 16nm technology.

For more in-depth analysis, the number of TSVs used in the critical paths are presented in Table 40. If the TSV count is zero, the critical path is a 2D path existing in a single die. If the TSV count is three, the critical path alternates three times (e.g., die0 – die1 – die0 – die1) between two dies since all the layouts are two-die designs. Especially, if the TSV count is zero and the critical path delay is shorter than the critical path delay of its 2D counterpart design, the shorter critical path delay of the 3D design is primarily due to the shorter wirelength achieved by the smaller footprint area. On the other hand, if the TSV count is non-zero, the critical path delay comes from both the smaller footprint area and the shorter wirelength.

6.4.5 Impact on Power

Figure 68 and Figure 69 show power consumption for BM1 and BM2 benchmark circuits, respectively. As seen in the figures, moving from 2D ICs to 3D ICs does not necessarily lead to power reduction even if 3D designs have shorter wirelength than 2D designs. The reason is as follows. Reduction in power consumption by building 3D ICs comes from smaller dynamic power consumption due to shorter wirelength.⁴ However, TSV capacitance can essentially be thought of as wire capacitance. Therefore, the total capacitance is the sum of the total TSV capacitance and the total wire capacitance. This means that the total TSV capacitance should be less than the reduced wire capacitance to achieve power reduction.⁵ In other words, achievement of power reduction needs smaller TSV capacitance, use of fewer TSVs, and wirelength reduction. However, there again exist trade-offs among the number of TSVs, the amount of wirelength reduction, and power consumption. Inserting fewer TSVs may not reduce the total wirelength as much as expected. Similarly,

⁴There exist many kinds of 3D integration and some of them (e.g., core-DRAM stacking) provide a huge amount of power saving by removing long chip-to-chip connections.

⁵Note that this is a simplified analysis. In reality, the total power should be computed in a more sophisticated fashion taking switching activities of nets and gates into account.

the use of fewer TSVs may not reduce the dynamic power consumption. Inserting more TSVs, however, may reduce the total wirelength more than 10% to 20% [11], but then the total TSV capacitance also increases, so the total capacitance could be larger than the total capacitance of 2D designs.

6.5 Summary

In this chapter, the impact of TSVs on the quality of today and future 3D ICs is investigated using GDSII-level layouts. To generate 3D IC layouts of future 3D IC layouts, $22nm$ and $16nm$ process and standard cell libraries are developed based on the ITRS prediction and downscaling trends of other standard cell libraries and Intel process technology. With these realistic libraries, today and future 3D IC layouts are generated and their quality is compared. The simulation results show that 1) footprint area is strongly dependent on the TSV size, 2) wirelength is also dependent on the TSV size, but if the TSV size is sufficiently small ($0.5\mu m$ TSVs for $16nm$ technology), further shrinking the TSV size does not help wirelength reduction, 3) critical path delay is strongly dependent on the TSV capacitance, but footprint area also has a non-negligible effect on critical path delay, and 4) transition from 2D ICs to 3D ICs does not necessarily lead to less power consumption even when the TSV capacitance is small.

CHAPTER 7

TOPOGRAPHY VARIATION IN 3D INTEGRATED CIRCUITS

Topography variation in metal layers is becoming more serious as technology advances beyond $65nm$ and $45nm$, and as a result, semiconductor manufacturers are putting tighter metal density rules in their design rule decks. Moreover, it is also required to minimize the range of metal density¹ and the maximum metal density gradient² because topography is determined mainly by underlying feature density [73, 74]. In addition, topography is cumulative so the flatter the topography in lower metal layers is, the better the topographies of upper metal layers are [75].

In order to improve metal density and achieve uniform density distribution, various design methodologies have been proposed. The authors of [76] proposed fill insertion as a post-routing process. The authors of [77] addressed the metal density problem in global routing. CMP-aware placement was also proposed in [78]. Among all of these techniques, fill insertion has been widely used to achieve uniform density distribution. During fill insertion, *fills* (dummy metal pieces) are inserted into whitespace in order to not only satisfy metal density constraints but also improve related density metrics.

Meanwhile, three-dimensional integrated circuits (3D ICs) have emerged to resolve the interconnect bottleneck and improve performance of 2D ICs further. In 3D ICs, cells are placed in multiple dies, the dies are stacked vertically, and through-silicon vias (TSVs) are used to connect metal layers of adjacent dies as shown in Figure 70. Since footprint area of 3D ICs becomes smaller than that of 2D ICs, the total wirelength becomes shorter than 2D ICs, so it is expected that the performance of 3D ICs is better than 2D ICs [2, 11, 47].

Via-first-type TSVs, however, are attached to landing pads in the bottommost and the topmost metal layers as shown in Figure 70. These metal landing pads are usually very

¹Range of metal density is defined as the difference between the maximum density and the minimum density.

²Metal density gradient is defined as the density difference between two adjacent windows.

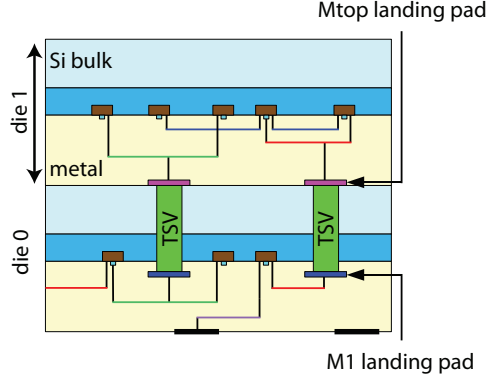


Figure 70. 3D IC designed in two dies (left) and three dies (right) using via-first TSVs and face-to-back die bonding.

large (see Figure 71), so they cause significant metal density mismatch which will be shown in the next section.

In this chapter, a 3D global placement algorithm is extended to improve metal1 density in 3D ICs. This algorithm improves the range of metal density as well as the maximum density gradient significantly compared to traditional wirelength-driven placement. In addition, the impact of the landing size on metal density metrics is also investigated.

7.1 Motivation

7.1.1 Feature Density of 2D and 3D IC Layouts

As mentioned in the previous section, metal1 landing pads are much bigger than the minimum feature size. Figure 71 shows an example. The landing pad width in the figure is $4.14\mu m$ but the minimum width of metal1 wire is $65nm$ which is approximately $63\times$ smaller. Therefore metal1 density of layout regions containing landing pads is much higher than other layout regions devoid of landing pads.

To investigate density variations caused by landing pads further, a preliminary simulation is performed on a 2D layout ($1300\mu m \times 1300\mu m$). In this simulation, landing pads are inserted only in one window area ((0,0) to $(100\mu m, 100\mu m)$) and metal densities before and after fill insertion are compared. Figure 72 shows the result. When the number of landing pads is less than 30 to 50, the maximum density window is different from the TSV

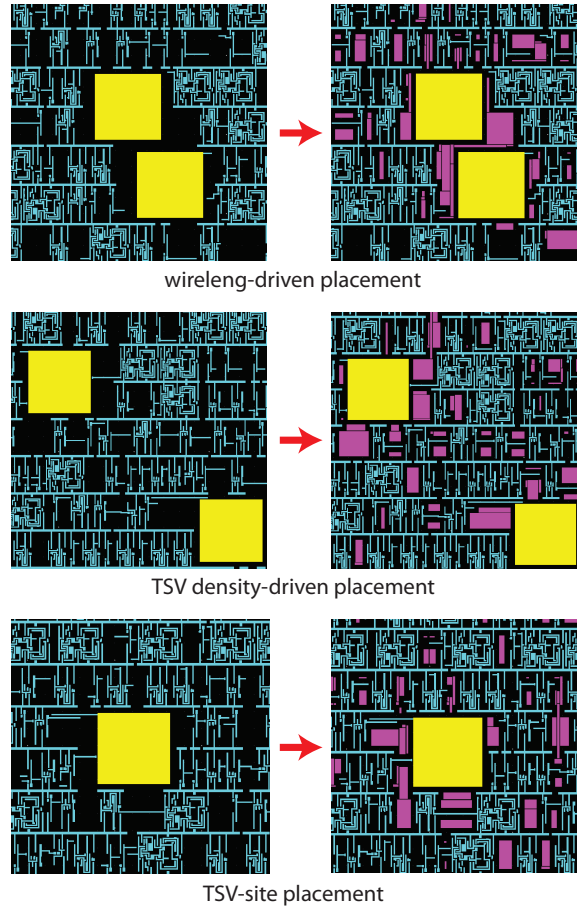


Figure 71. Before and after filler insertion. Yellow squares denotes TSVs, pink are fillers, and light blue are M1 wires.

window (D_{TSV}) in which landing pads exist, so it is relatively easy to control the density range over the entire layout by fill insertion. However, as the number of landing pads in the TSV window increases, the TSV window becomes the maximum density window, and the density range increases almost linearly as the landing pad increases. Therefore, it is necessary to keep the number of landing pads in one window small or spread landing pads well.

7.1.2 The Design of 3D ICs

The authors of [11] have proposed two 3D IC design schemes, namely TSV co-placement and TSV-site. In TSV co-placement scheme, they place TSVs and cells simultaneously so

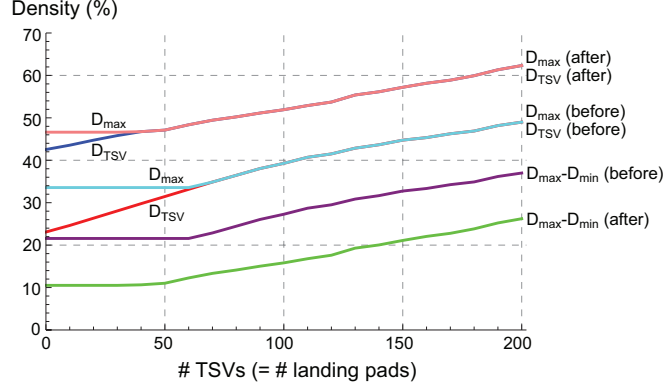


Figure 72. Variations of the maximum density and the density range in metal1 layer when only one window contains landing pads ($4.14\mu\text{m} \times 4.14\mu\text{m}$). ‘before’ (or ‘after’) denotes ‘before’ (or ‘after’) fill insertion, D_{\max} (or D_{\min}) denotes the maximum (or minimum) window density, and D_{TSV} denotes the density of the window containing landing pads.

that they can minimize the total wirelength consisting of wirelength of cell-to-cell connections, wirelength of cell-to-TSV connections, and wirelength of TSV-to-TSV connections. In TSV-site scheme, on the other hand, they place TSVs uniformly on the entire layout area and then place cells. In this case, they need to assign 3D nets to TSVs to determine which 3D net uses which TSV. Since the solution set of TSV co-placement scheme contains that of TSV-site scheme, wirelength of TSV co-placement is shorter than TSV-site scheme. However, it is expected that TSV-site scheme will have better metal1 density.

In this chapter, simulations on these two design schemes are included too because they are two extreme placement schemes and our TSV density-driven placement algorithm stands between them.

7.2 TSV Density-Driven 3D Global Placement

The 3D placement algorithm used in this work is based on the force-directed quadratic placement algorithm for 3D ICs [73, 11]. In this section, therefore, the force-directed quadratic 3D placement algorithm is reviewed and how it is extended for TSV density-driven 3D placement is presented.

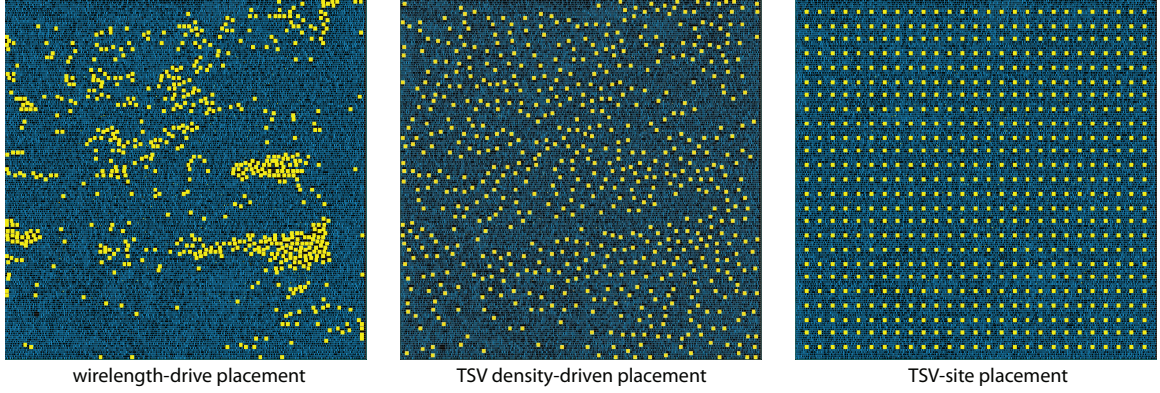


Figure 73. Screen shots of die0 of circuit C2. Dark rectangles are standard cells, and light squares are metal1 landing pads.

7.2.1 Force-Directed Quadratic 3D Global Placement

The basic principle of force-directed quadratic 2D global placement is to apply several forces to cells and pins, and move cells gradually until the cell occupancy of each global bin becomes less than a pre-determined number. When the forces are applied, objective functions such as quadratic wirelength are minimized.

The authors of [54] suggested three forces for the force-directed placement. The first force is *net force* which pulls connected cells so that it minimizes the total wirelength. The second force is *move force* which spreads cells out so that it removes cell overlaps. The third force is *hold force* which holds cells at the current location so that cells in low cell density regions do not move. The sum of the forces is set to zero to minimize wirelength while removing cell overlaps. This is mathematically expressed as follows:

$$\mathbf{f} = \mathbf{f}^{\text{net}} + \mathbf{f}^{\text{move}} + \mathbf{f}^{\text{hold}} = \mathbf{0} \quad (80)$$

where \mathbf{f}^{net} , \mathbf{f}^{move} , and \mathbf{f}^{hold} are net force, move force, and hold force respectively.

The authors of [11] extended this algorithm so that they can place cells in 3D. They first use multi-way partitioning to split cells into multiple partitions (dies in 3D ICs). After partitioning, they place cells and TSVs with the same objective function as 2D placement. However, they compute all the forces for each die separately because cells in different dies do not overlap.

7.2.2 TSV Density-Driven 3D Global Placement

Since closely-placed TSVs can cause serious density mismatch in metal1 layer, another density force focusing on TSVs only, namely TSV density force, is applied. This force is similar as the move force (\mathbf{f}^{move}) which is actually computed by cell density. TSV density force is computed as explained below. First, placement density considering TSV density only in each bin is computed as follows:

$$D(b)\Big|_{z=d} = D^{\text{TSV}}(b)\Big|_{z=d} - D^{\text{chip}}(b)\Big|_{z=d} \quad (81)$$

where $D^{\text{TSV}}(b)\Big|_{z=d}$ is the TSV density in the bin b of d -th die, and $D^{\text{chip}}(b)\Big|_{z=d}$ is the total capacity of bin b of d -th die. Then, the placement potential Φ^{TSV} is computed by Poisson equation:

$$\Delta \Phi^{\text{TSV}}(b)\Big|_{z=d} = -D(b)\Big|_{z=d} \quad (82)$$

Then, the x-location of i -th TSV in the next iteration is computed by the following equation:

$$x_i = x'_i - \frac{\partial}{\partial x} \Phi^{\text{TSV}}(b)\Big|_{(b'), z=d} \quad (83)$$

where x_i is the target x-location, x'_i is the current x-location, and b' is the current bin in which i -th TSV exists. y-location of i -th TSV is computed in the similar way.

If the TSV density force is computed as above, the final force equation becomes as follows:

$$\mathbf{f} = \mathbf{f}^{\text{net}} + \mathbf{f}^{\text{move}} + \mathbf{f}^{\text{hold}} + \mathbf{f}^{\text{TSV}} = \mathbf{0} \quad (84)$$

where \mathbf{f}^{TSV} is the TSV density force.

Since TSVs are treated as cells during 3D placement to find the optimal locations of TSVs, \mathbf{f}^{move} and \mathbf{f}^{hold} include TSV density, which is also included in \mathbf{f}^{TSV} , as well as cell density during cell density computation. Therefore, several sub-options such as 1) include TSV and cell densities in \mathbf{f}^{move} and \mathbf{f}^{hold} , and 2) include cell density only in \mathbf{f}^{move} but include TSV and cell densities in \mathbf{f}^{hold} exist. In order to obtain the best results, all these options were tried and it was found that including both TSV and cell densities in both \mathbf{f}^{move} and

Table 41. Benchmark Circuits. The number of TSVs is based on two-die implementation.

ckt	# gates	# nets	# TSVs	# TSVs / # nets
C1	29,706	29,979	1,035	0.0345
C2	77,234	77,378	675	0.0087
C3	88,401	89,149	1,045	0.0117
C4	103,711	103,946	424	0.0041
C5	109,181	109,415	1745	0.0159
C6	168,943	169,469	114	0.0007
C7	324,490	327,843	1559	0.0048
C8	444,555	483,563	3838	0.0079

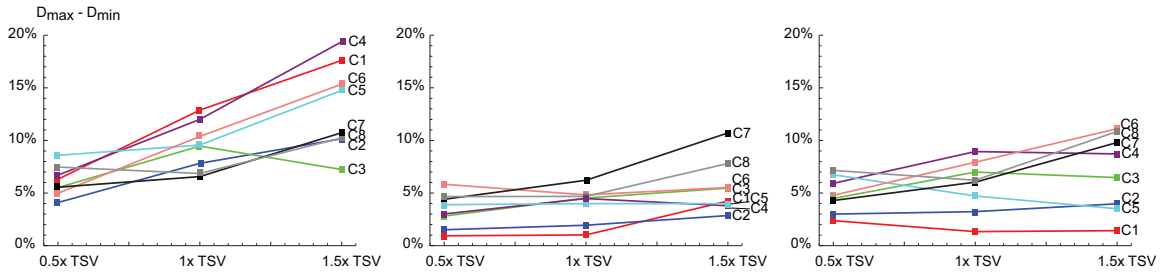


Figure 74. ΔD of die0 of WL-driven placement (left), TSV-site placement (middle), and TSV density-driven placement (right)

f^{hold} generated the best results. This observation is also intuitively understood because 1) if only the cell density is included in f^{move} , the cell occupancy of a bin fully occupied by cells and TSVs cannot be reduced, so routing may fail or overlap among cells and TSVs cannot be removed effectively, and 2) f^{hold} should be balanced with f^{move} when the density of a bin is sufficiently low, so if f^{move} considers cells and TSVs, f^{hold} should also consider cells and TSVs during density computation.

For better understanding, die0 (the bottommost die in Figure 70) layouts of circuit C2 designed by wirelength-driven placement, TSV density-driven placement, and TSV-site placement are shown in Figure 73. The left figure shows the WL-driven placement, so the TSVs are placed non-uniformly. On the other hand, the middle figure shows the TSV density-driven placement, so the TSVs are sparsely placed. The right figure shows uniformly-placed TSVs in TSV-site placement.

Table 42. Multi-pass metal1 fill insertion parameters.

Parameter	Value
Min. fill-to-object distance	0.325 μm
The amount of decrement in fill width	0.065 μm
Offset for staggering	0.130 μm
Min. fill-to-fill distance	0.065 μm
Max. metal density	75%
Max. length (or width) of a metal fill	3.25 μm
Min. metal density	25%
Min. length (or width) of a metal fill	0.065 μm
Preferred metal density	35%
Window size (width)	100 μm
Window step size	50 μm

Table 43. Comparison of minimum and maximum metal1 densities in two-die implementation with 1 \times TSV. ‘before (or after)’ denotes ‘before (or after) fill insertion’.

ckt	Die	Minimum density						Maximum density					
		WL-driven		TSV-site		TSV density-driven		WL-driven		TSV-site		TSV density-driven	
		before	after	before	after	before	after	before	after	before	after	before	after
C1	die0	32.823%	34.044%	39.770%	39.770%	41.880%	41.880%	46.926%	46.926%	43.991%	43.991%	43.215%	43.215%
	die1	18.431%	38.553%	18.498%	36.864%	18.498%	37.611%	18.977%	40.118%	19.321%	39.046%	19.088%	38.633%
C2	die0	27.157%	34.978%	29.548%	38.412%	29.294%	38.429%	35.725%	42.803%	31.304%	41.152%	31.253%	41.664%
	die1	22.899%	37.763%	22.999%	37.429%	22.888%	37.837%	23.503%	39.582%	23.613%	39.647%	23.547%	39.779%
C3	die0	20.318%	35.092%	22.795%	39.471%	21.879%	37.162%	36.502%	44.556%	26.605%	43.752%	27.203%	44.158%
	die1	16.962%	39.586%	16.967%	39.219%	16.917%	39.407%	19.910%	44.267%	19.943%	42.986%	19.903%	43.935%
C4	die0	23.610%	31.291%	25.447%	33.160%	24.990%	32.171%	36.396%	43.292%	28.689%	39.101%	29.354%	41.114%
	die1	22.605%	32.130%	22.728%	31.915%	22.535%	31.828%	25.092%	36.524%	25.189%	37.922%	25.182%	36.305%
C5	die0	32.884%	34.661%	34.597%	39.463%	35.956%	37.858%	43.832%	44.224%	39.738%	41.646%	39.289%	42.580%
	die1	22.503%	33.391%	22.657%	33.203%	22.738%	33.398%	24.510%	37.418%	24.737%	36.576%	24.369%	37.364%
C6	die0	23.276%	32.209%	23.438%	32.279%	23.265%	32.566%	29.103%	42.590%	26.434%	37.755%	21.717%	33.671%
	die1	21.658%	33.616%	21.703%	32.505%	21.717%	33.671%	25.339%	38.298%	25.088%	38.124%	25.373%	38.494%
C7	die0	18.853%	40.446%	14.470%	40.454%	17.596%	41.140%	32.825%	47.010%	17.687%	50.431%	23.007%	47.173%
	die1	16.226%	42.194%	13.466%	42.079%	15.367%	42.012%	18.751%	47.370%	16.208%	49.616%	17.914%	48.227%
C8	die0	23.708%	39.600%	24.987%	41.176%	24.279%	39.599%	36.249%	46.460%	27.616%	46.641%	28.207%	45.818%
	die1	21.384%	40.188%	21.408%	39.905%	21.417%	40.052%	22.909%	44.984%	22.984%	45.032%	22.839%	44.734%
Geo. mean	die0	24.869%	35.165%	25.858%	37.884%	26.452%	33.445%	36.823%	44.700%	29.274%	42.891%	29.649%	42.238%
	die1	20.167%	37.016%	19.761%	36.470%	20.061%	36.819%	36.470%	40.906%	21.904%	40.921%	22.104%	40.752%

7.3 Simulation Results

Eight benchmark circuits obtained from IWLS 2005 benchmark suite [29] and Open-Cores [79] are used in the simulation. These circuits are listed in Table 41. NCSU 45nm technology library [80] is also used. The baseline TSV landing pad size (1 \times TSV) is 4.14 $\mu m \times$ 4.14 μm , and Table 42 shows our fill insertion parameters.

Table 44. Comparison of wirelength and metal1 densities in two-die implementation with $1\times$ TSV. D denotes metal1 density of a window. (Numbers in parentheses are wirelength ratios.)

ckt	Wirelength (mm)			Die	Range ($\Delta D = D_{\max} - D_{\min}$)			Maximum density gradient		
	WL-driven	TSV-site	TSV density-driven		WL-driven	TSV-site	TSV density-driven	WL-driven	TSV-site	TSV density-driven
C1	0.783	0.861	0.812	die0	12.882%	4.221%	1.335%	6.986%	2.906%	0.642%
	(1.000)	(1.100)	(1.037)	die1	1.566%	2.182%	1.021%	1.040%	0.982%	0.588%
C2	1.680	1.735	1.718	die0	7.825%	2.739%	3.234%	5.029%	1.194%	1.808%
	(1.000)	(1.033)	(1.023)	die1	1.819%	2.219%	1.942%	1.577%	1.949%	1.562%
C3	2.468	2.595	2.558	die0	9.465%	4.528%	6.995%	5.794%	2.121%	3.118%
	(1.000)	(1.051)	(1.036)	die1	4.681%	3.767%	4.528%	2.273%	1.799%	1.895%
C4	2.385	2.441	2.456	die0	12.002%	5.941%	8.943%	5.778%	3.021%	4.240%
	(1.000)	(1.023)	(1.030)	die1	4.395%	6.001%	4.476%	1.844%	2.733%	2.230%
C5	2.328	2.482	2.413	die0	9.563%	2.184%	4.722%	4.258%	1.341%	2.274%
	(1.000)	(1.066)	(1.037)	die1	4.027%	3.373%	3.966%	2.178%	3.137%	2.417%
C6	3.925	3.961	3.881	die0	10.381%	5.476%	7.930%	5.883%	2.173%	4.049%
	(1.000)	(1.010)	(0.989)	die1	4.682%	5.619%	4.824%	2.525%	2.236%	2.055%
C7	13.744	15.582	14.050	die0	6.564%	9.978%	6.033%	4.134%	7.627%	3.303%
	(1.000)	(1.134)	(1.022)	die1	5.176%	7.537%	6.215%	3.576%	5.317%	3.303%
C8	15.410	16.595	15.599	die0	6.860%	5.465%	6.219%	4.797%	2.229%	3.636%
	(1.000)	(1.077)	(1.012)	die1	4.796%	5.126%	4.682%	2.436%	3.194%	2.247%
Geo. mean	3.326	3.529	3.403	die0	9.197%	4.574%	4.982%	5.258%	2.400%	2.506%
	(1.000)	(1.061)	(1.023)	die1	3.587%	4.102%	3.497%	2.064%	2.405%	1.832%

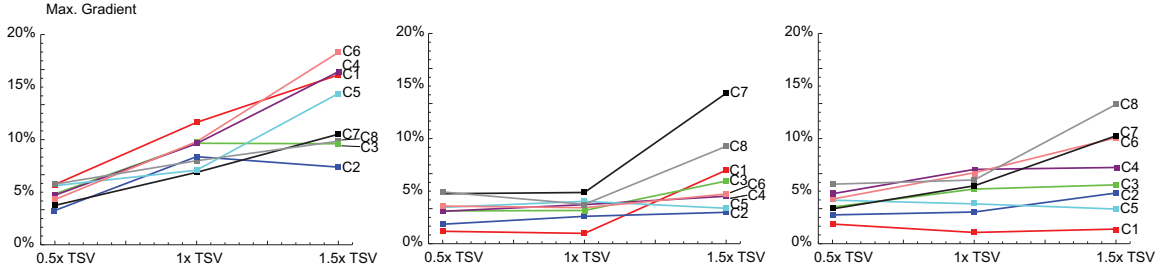


Figure 75. Maximum density gradient of die0 of WL-driven placement (left), TSV-site placement (middle), and TSV density-driven placement (right)

7.3.1 Metal1 Density Comparison

The first comparison is on the minimum and the maximum metal1 densities before and after fill insertion to show that the fill insertion tool satisfies the lower (25%) and the upper (75%) limits of metal densities and achieves the preferred density (35%). Table 43 shows the results. As all the ‘after’ columns show, the fill insertion tool satisfies the metal density limits well for both die0 in which TSVs exist and die1 in which TSVs do not exist. Moreover, final metal densities are close to the preferred metal density (35%). From this table, it is observed that metal densities of 3D IC layouts can satisfy lower and upper density limits after fill insertion even when large landing pads exist. In addition, metal densities after fill insertion even for the two extreme TSV placement cases (WL-driven placement and TSV-site placement) satisfy the minimum and the maximum density requirements.

Next, metal1 densities of all the benchmark circuits designed in two dies with $1\times$ TSV

Table 45. Critical path delay, power, and the number of fills.

ckt	Critical path delay (ns)			Power (W)			# fills			
	WL-driven	TSV-site	TSV density-driven	WL-driven	TSV-site	TSV density-driven	Die	WL-driven	TSV-site	TSV density-driven
C1	6.10	5.68	5.51	0.0343	0.0345	0.0342	die0	2,537	0	0
							die1	22,465	24,750	25,399
C2	4.80	5.15	4.65	0.166	0.165	0.166	die0	33,351	36,056	33,150
							die1	41,404	41,043	40,842
C3	4.48	4.49	4.44	0.121	0.135	0.119	die0	57,139	64,332	60,792
							die1	80,152	81,760	82,192
C4	2.29	2.88	2.72	0.165	0.165	0.165	die0	42,561	46,677	44,654
							die1	49,032	49,580	49,116
C5	1.90	2.13	1.78	0.371	0.368	0.367	die0	15,397	19,674	16,740
							die1	55,480	57,766	56,815
C6	2.84	3.85	3.06	0.284	0.285	0.284	die0	77,774	76,574	78,231
							die1	83,204	84,069	84,654
C7	6.41	5.98	5.27	1.194	1.120	1.196	die0	327,859	390,706	342,453
							die1	342,065	396,116	357,603
C8	63.31	62.22	63.46	2.508	2.590	2.626	die0	254,218	270,061	265,014
							die1	306,681	324,871	317,980

are compared. In this two-die implementation, *die0* contains TSVs as well as cells but *die1* contains only cells as shown in Figure 70. Table 44 shows the density results.

Comparing ΔD which is the difference between the maximum density and the minimum density, it is observed that WL-driven placement has the worst density range compared to TSV density-driven placement or TSV-site placement in *die0*. The geometric mean of ΔD of WL-driven placement is about 9.197% whereas that of TSV density-driven placement is 4.982% and that of TSV-site placement 4.574%. Similarly, the maximum gradient, which is the maximum difference between densities of two adjacent windows, of WL-driven is worse than TSV density-driven or TSV-site placement. The geometric mean of the maximum density gradient of WL-driven placement is 5.258% but that of TSV density-driven placement is 2.506% and that of TSV-site placement is 2.400% in *die0*. Therefore, uniformly-placed TSVs improve metal1 densities significantly.

However, metal1 density in *die1* shows different trends because *die1* does not contain landing pads in metal1 layer. As the table shows, the geometric mean of density range or the maximum density gradient of WL-driven placement is similar to that of TSV density-driven or TSV-site placement. Therefore, TSV density-driven placement (or TSV-site placement) achieves better metal1 density than WL-driven placement when landing pads exist and similar metal1 density as WL-driven placement when landing pads do not exist.

The reason that ΔD of TSV density-driven placement is similar as that of WL-driven

placement for big circuits such as C7 and C8 is that few TSVs are spread over large layout area, so the impact of landing pads on ΔD becomes smaller. On the other hand, if there are many TSVs (C1 and C5) compared to its layout area, TSV density-driven placement outperforms WL-driven placement with respect to metal1 density. In addition, TSV-site shows the best ΔD results, but ΔD of TSV density-driven placement is close to that of TSV-site placement.

7.3.2 Wirelength Comparison

WL-driven placement has three basic forces (net force, hold force, and move force). However, one more force is added in TSV density-driven placement, and TSVs are pre-placed uniformly in TSV-site placement, thus the wirelength of TSV density-driven placement or TSV-site placement is expected to be longer than WL-driven placement. Table 44 also shows wirelength comparison. The average wirelength of WL-driven placement is $3.326mm$ while that of TSV density-driven placement is $3.403mm$ which is 2.3% longer than WL-driven placement. On the other hand, the average wirelength of TSV-site placement is $3.529mm$ which is 6.1% longer than WL-driven placement. Therefore, TSV density-driven placement improves metal1 density significantly (approximately two times better than WL-driven placement with respect to both ΔD and the maximum gradient, and very comparable to TSV-site placement) with just 2.3% wirelength overhead. Moreover, wirelength overhead of TSV density-driven placement remains between 1.2% and 3.7% but the improvement in metal density is huge ($2\times$ to $9\times$). On the other hand, wirelength overhead of TSV-site placement is between 1.0% and 13.4% which is much worse than TSV density-driven placement. As a result, TSV density-driven placement is comparable to TSV-site placement with respect to metal1 density and comparable to WL-driven placement with respect to wirelength.

7.3.3 Impact of Landing Pad Size

Since the landing pad size affects metal1 density significantly, the impact of landing pad size on density metrics is also investigated. Figure 74 shows ΔD for all the circuits when the landing pad size is $0.5 \times (2.07 \times 2.07 \mu m^2)$, $1 \times (4.14 \times 4.14 \mu m^2)$, and $1.5 \times (6.21 \times 6.21 \mu m^2)$. In general, ΔD increases as the landing pad size goes up in WL-driven placement. However, ΔD decreases in some cases as the landing pad size increases as shown in TSV density-driven or TSV-site placement cases. Therefore, larger TSV landing pad size does not always lead to worse ΔD . This is mainly because fill insertion can somehow increase the minimum density to decrease ΔD if TSVs are spread out sufficiently. Similarly, maximum density gradient does not always increase as the landing pad size goes up as shown in Figure 75.

7.3.4 Timing and Power Comparison

Timing and power analysis is conducted using Synopsys PrimeTime and the results are shown in Table 45. Critical path delay of WL-driven placement is smaller than other two placements in C4, C5, and C6. On the other hand, TSV density-driven placement has smaller critical path delay than other two placements in C1, C2, C3, C5, and C7. Since all the placement algorithms are not timing-driven, TSV density-driven placement is not always better than WL-driven or TSV-site placement with respect to timing, but 2.3% wirelength overhead of TSV density-driven placement does not lead to worse critical path delay.

On the other hand, power is almost same for all the three placement algorithms. Since gate delays are almost similar for all of them, the additional power consumption comes from interconnect power. However, wirelength overhead of TSV density-driven or TSV-site placement is approximately 1% to 13%, so the total power consumption is almost same for all the cases.

7.3.5 Number of Fills

The number of fills is also an important metric for fill insertion because too many fills inserted in a design can increase the data volume and RC extraction time significantly. Therefore, the number of fills are reported in Table 45.

The fill counts for all three placement styles are almost same except C1 as shown in the table. In case of die0 of C1, the minimum and maximum densities already satisfy the density requirements so the fill insertion tool does not insert any fills.

7.4 Summary

In this chapter, topography variation of 3D ICs is investigated. TSV landing pads are typically large, so TSVs inserted inside the core area could result in serious metal density mismatch. In order to reduce topography variation in 3D ICs, a 3D global placement algorithm is extended to a TSV density-driven 3D global placement algorithm. In the algorithm, a new force acting only on TSVs is added to spread TSVs out with little wirelength overhead. In the simulation results, $1.86\times$ improvement in the range of metal1 density and $2.10\times$ improvement in the maximum metal1 density gradient are achieved compared to wirelength-driven placement. Wirelength overhead of the extended placement algorithm is just 2.3%, which is almost negligible. On the other hand, wirelength overhead of the TSV-site placement is much higher than the TSV density-driven placement. Therefore, TSV density-driven placement achieves short wirelength comparable to wirelength-driven placement and small metal density variation comparable to TSV-site placement. The impact of landing pad size on metal1 density is also presented. The metal1 density range and the maximum density gradient of wirelength-driven placement become worse as the landing pad size increases. Those of TSV density-driven placement also increase as the landing pad size increased, but this is observed only when few TSVs exist in the layouts. In summary, the TSV density-driven placement has much better metal1 density characteristics than the wirelength-driven placement.

CHAPTER 8

CONCLUSIONS

Three-dimensional integrated circuits (3D ICs) are expected to be a breakthrough technology for high performance computing, heterogeneous integration, low power ICs, extremely small devices, and so on. Since previous prediction models and design methodologies and algorithms for 3D ICs have not taken signal through-silicon vias (TSVs) into account, this work has developed more accurate prediction models for 3D ICs, and proposed and implemented design methodologies and algorithms that take TSVs into account. This thesis presents the following:

- A TSV-aware wirelength, delay, and power prediction model for gate-level 3D ICs.
- A TSV-aware wirelength prediction model for block-level 3D ICs.
- Analytical models of TSV capacitive coupling in 3D ICs.
- Design methodologies and algorithms for gate-level 3D ICs.
- Design methodologies and algorithms for block-level 3D ICs.
- A study on the impact of TSVs on the quality of 3D ICs.
- Topography variation in 3D ICs.

The TSV-aware interconnect prediction model presented in this dissertation is more accurate than other prediction models and relates area, wirelength, TSV count, and TSV size. The analytical models of TSV capacitive coupling provide fast estimation of TSV coupling capacitance. Since the computation time of these models is almost negligible, the models are suitable for use in design steps such as floorplanning and global placement, both of which require fast estimation of TSV capacitances for rough timing optimization. The

design methodologies and algorithms for gate- and block-level 3D ICs show that they generate DRC-clean 3D IC layouts with a reasonable number of TSVs. The study pertaining to the impact of TSVs on the quality of 3D ICs investigates and compares the quality of 3D ICs built with various device and TSV technologies. It also provides guidelines on the maximum TSV size and capacitance for each device technology. The study of topography variation presents the impact of metal landing pads on the topography variation of 3D ICs. To minimize such variation, a technique applied to the force-directed quadratic placement algorithm is also proposed.

Despite the many contributions of this research, its limitations that call for further investigation must be addressed. For one, the interconnect prediction model assumes that TSVs are placed uniformly on the layout. Since TSVs can also be placed non-uniformly, more accurate prediction models should take non-uniformly-placed TSVs into account. In addition, more realistic buffer insertion methodologies such as dynamic programming-based buffer insertion algorithms should be used to predict delay and power more accurately. Since the TSV coupling capacitance models are not highly accurate, to decrease error, the models must be improved. Although current methodologies and algorithms for the design of gate- and block-level 3D ICs are workable, they must be more sophisticated and effective. For instance, the block-level 3D IC design methodology estimates TSV locations regardless of the existing whitespace locations, and then assigns the estimated TSV locations to nearby whitespace. Therefore, TSV insertion algorithms that account for existing whitespace may be more effective than the TSV insertion algorithms presented in this dissertation.

Future research pertaining to 3D ICs could follow several interesting directions. Although nearly all of the work in this dissertation uses via-first TSVs, via-last TSVs can also be used to build 3D ICs. If fabricating different types of TSVs in a single die becomes possible, it would be interesting to develop design methodologies and algorithms for the simultaneous planning of different types of TSVs in the design of 3D ICs. 3D placement,

3D routing, and TSV insertion are also challenging research areas. One important research question is when TSVs should be inserted because it relates to design algorithms such as 3D placement and routing as well as design methodologies. With the development of effective design methodologies and physical design algorithms, 3D ICs should provide much higher bandwidth, improved performance, lower power, and a smaller form factor than 2D ICs.

REFERENCES

- [1] Michael B. Healy and Krit Athikulwongse and Rohan Goel and Mohammad M. Hossain and Dae Hyun Kim and Young-Joon Lee and Dean L. Lewis and Tzu-Wei Lin and Chang Liu and Moongon Jung and Brian Ouellette and Mohit Pathak and Hemane Sane and Guanhao Shen and Dong Hyuk Woo and Xin Zhao and Gabriel H. Loh and Hsien-Hsin S. Lee and Sung Kyu Lim, “Design and Analysis of 3D-MAPS: A Many-core 3D Processor with Stacked Memory,” in *Proceedings IEEE Custom Integrated Circuits Conference*, Oct. 2010.
- [2] T. Thorolfsson, K. Gonsalves, and P. D. Franzon, “Design Automation for a 3DIC FFT Processor for Synthetic Aperture Radar: A Case Study,” in *Proceedings IEEE/ACM Design Automation Conference*, pp. 51–56, July 2009.
- [3] M.-C. Tsai, T.-C. Wang, and T. Hwang, “Through-Silicon Via Planning in 3-D Floorplanning,” in *IEEE Transactions on VLSI Systems*, 2010.
- [4] P. Bai *et al.*, “A 65nm Logic Technology Featuring 35nm Gate Lengths, Enhanced Channel Strain, 8 Cu Interconnect Layers, Low-k ILD and 0.57 μm^2 SRAM Cell,” in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2004.
- [5] K. Mistry *et al.*, “A 45nm Logic Technology with High-k+Metal Gate Transistors, Strained Silicon, 9 Cu Interconnect Layers, 193nm Dry Patterning, and 100% Pb-free Packaging,” in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2007.
- [6] P. Packan *et al.*, “High Performance 32nm Logic Technology Featuring 2nd Generation High-k + Metal Gate Transistors,” in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2009.
- [7] D. H. Kim, K. Athikulwongse, M. B. Healy, M. M. Hossain, M. Jung, I. Khorosh, G. Kumar, Y.-J. Lee, D. L. Lewis, T.-W. Lin, C. Liu, S. Panth, M. Pathak, M. Ren, G. Shen, T. Song, D. H. Woo, X. Zhao, J. Kim, H. Choi, G. H. Loh, H.-H. S. Lee, and S. K. Lim, “3D-MAPS: 3D Massively Parallel Processor with Stacked Memory,” in *Int. Solid-State Circuits Conference*, Feb. 2012.
- [8] D. H. Kim, S. Mukhopadhyay, and S. K. Lim, “Fast and Accurate Analytical Modeling of Through-Silicon-Via Capacitive Coupling,” in *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, vol. 1, pp. 168–180, Jan. 2011.
- [9] J. W. Joyner, P. Zarkesh-Ha, J. A. Davis, and J. D. Meindl, “A Three-Dimensional Stochastic Wire-Length Distribution for Variable Separation of Strata,” in *Proceedings IEEE International Interconnect Technology Conference*, pp. 126–128, June 2000.

- [10] X. Dong and Y. Xie, "System-Level Cost Analysis and Design Exploration for Three-Dimensional Integrated Circuits (3D ICs)," in *Proceedings Asia and South Pacific Design Automation Conference*, pp. 234–241, Jan. 2009.
- [11] D. H. Kim, K. Athikulwongse, and S. K. Lim, "A Study of Through-Silicon-Via Impact on the 3D Stacked IC Layout," in *Proceedings IEEE/ACM International Conference on Computer-Aided Design*, pp. 674–680, Nov. 2009.
- [12] Nangate, "Nangate FreePDK45 Open Cell Library." <http://www.nangate.com>.
- [13] J.-S. Kim, C. S. Oh, H. Lee, D. Lee, H. R. Hwang, S. Hwang, B. Na, J. Moon, J.-G. Kim, H. Park, J.-W. Ryu, K. Park, S. K. Kang, S.-Y. Kim, H. Kim, J.-M. Bang, H. Cho, M. Jang, C. Han, J.-B. Lee, J. S. Choi, and Y.-H. Jun, "A 1.2 V 12.8 GB/s 2 Gb Mobile Wide-I/O DRAM with 4×128 I/Os Using TSV Based Stacking," in *IEEE Journal of Solid-State Circuits*, pp. 107–116, Jan. 2012.
- [14] R. Zhang, K. Roy, C.-K. Koh, and D. B. Janes, "Stochastic Wire-Length and Delay Distributions of 3-Dimensional Circuits," in *Proceedings IEEE/ACM International Conference on Computer-Aided Design*, pp. 208–213, Nov. 2000.
- [15] M. Lin, J. Luo, and Y. Ma, "A Low-Power Monolithically Stacked 3D-TCAM," in *Proceedings IEEE International Symposium on Circuits and Systems*, pp. 3318–3321, May 2008.
- [16] X. Zhao, J. Minz, and S. K. Lim, "Low-Power and Reliable Clock Network Design for Through-Silicon Via (TSV) Based 3D ICs," in *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, vol. 1, pp. 247–259, Feb. 2011.
- [17] J. Kim, J. Cho, J. S. Pak, T. Song, J. Kim, H. Lee, J. Lee, and K. Park, "I/O Power Estimation and Analysis of High-speed Channels in Through-Silicon Via (TSV)-based 3D IC," in *Proceedings IEEE Conference on Electrical Performance of Electronic Packaging and Systems*, pp. 41–44, 2010.
- [18] K-W Lee and A. Noriki and K. Kiyoyama and S. Kanno and R. Kobayashi and W-C Jeong and J-C Bea and T. Fukushima and T. Tanaka and M. Koyanagi, "3D Heterogeneous Opto-Electronic Integration Technology for System-on-Silicon (SOS)," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2009.
- [19] J. A. Davis, V. K. De, and J. D. Meindl, "A Stochastic Wire-Length Distribution for Gigascale Integration (GSI)-Part I: Derivation and Validation," in *IEEE Transactions on Electron Devices*, vol. 45, pp. 580–589, Mar. 1998.
- [20] J. W. Joyner, *Opportunities and Limitations of Three-dimensional Integration for Interconnect Design*. PhD thesis, Georgia Institute of Technology, 2003.
- [21] D. C. Sekar, A. Naeemi, R. Sarvari, J. A. Davis, and J. D. Meindl, "IntSim: A CAD tool for Optimization of Multilevel Interconnect Networks," in *Proceedings IEEE/ACM International Conference on Computer-Aided Design*, pp. 560–567, Nov. 2007.

- [22] J. U. Knickerbocker, P. S. Andry, L. P. Buchwalter, A. Deutsch, R. R. Horton, K. A. Jenkins, Y. H. Kwark, G. McVicker, C. S. Patel, R. J. Polastre, C. D. Schuster, A. Sharma, S. M. Sri-Jayantha, C. W. Surovic, C. K. Tsang, B. C. Webb, S. L. Wright, S. R. McKnight, E. J. Sprogis, and B. Dang, "Development of next-generation system-on-package (SOP) technology based on silicon carriers with fine-pitch chip interconnection," in *IBM J. Res. Dev.* 49(4/5), pp. 725–753, 2005.
- [23] J. U. Knickerbocker, P. S. Andry, B. Dang, R. R. Horton, M. J. Interrante, C. S. Patel, R. J. Polastre, K. Sakuma, R. Sirdeshmukh, E. J. Sprogis, S. M. Sri-Jayantha, A. M. Stephens, A. W. Topol, C. K. Tsang, B. C. Webb, and S. L. Wright, "Three-dimensional silicon integration," in *IBM J. Res. Dev.* 52(6), pp. 553–569, 2008.
- [24] M. Y. Lanzerotti, G. Fiorenza, and R. A. Rand, "Interpretation of Rent's Rule for Ultralarge-Scale Integrated Circuit Designs, With an Application to Wirelength Distribution Models," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 12, pp. 1330–1347, Dec. 2004.
- [25] B. S. Landman and R. L. Russo, "On a Pin Versus Block Relationship For Partitions of Logic Graphs," in *IEEE Transactions on Computers*, vol. c-20, pp. 1469–1479, Dec. 1971.
- [26] J. Cong, G. Luo, J. Wei, and Y. Zhang, "Thermal-Aware 3D IC Placement Via Transformation," in *Proceedings Asia and South Pacific Design Automation Conference*, pp. 780–785, Jan. 2007.
- [27] Synopsys, "Design Compiler." <http://www.synopsys.com>.
- [28] Cadence, "Soc Encounter." <http://www.cadence.com>.
- [29] IWLS, "IWLS 2005 Benchmarks." <http://www.iwls.org/iwls2005>.
- [30] G. Katti, M. Stucchi, K. D. Meyer, , and W. Dehaene, "Electrical Modeling and Characterization of Through Silicon via for Three-Dimensional ICs," in *IEEE Transactions on Electron Devices*, vol. 57, Jan. 2010.
- [31] Synopsys, "Raphael." <http://www.synopsys.com>.
- [32] D. H. Kim, S. Mukhopadhyay, and S. K. Lim, "TSV-aware Interconnect Length and Power Prediction for 3D Stacked ICs," in *Proceedings IEEE International Interconnect Technology Conference*, pp. 26–28, June 2009.
- [33] D. H. Kim, S. Mukhopadhyay, and S. K. Lim, "Through-Silicon-Via Aware Interconnect Prediction and Optimization for 3D Stacked ICs," in *Proc. ACM/IEEE Int. Workshop on System Level Interconnect Prediction*, pp. 85–92, July 2009.
- [34] R. Venkatesan, J. A. Davis, K. A. Bowman, and J. D. Meindl, "Optimal n -tier Multilevel Interconnect Architectures for Gigascale Integration (GSI)," in *IEEE Transactions on VLSI Systems*, no. 6, pp. 899–912, Dec. 2001.

- [35] E. Beyne, P. D. Moor, W. Ruythooren, R. Labie, A. Jourdain, H. Tilmans, D. S. Tezcan, P. Soussan, B. Swinnen, and R. Cartuyvels, "Through-Silicon Via and Die Stacking Technologies for Microsystems-integration," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2008.
- [36] F. Liu, R. R. Yu, A. M. Young, J. P. Doyle, X. Wang, L. Shi, K.-N. Chen, X. Li, D. A. Dipaola, D. Brown, C. T. Ryan, J. A. Hagan, K. H. Wong, M. Lu, X. Gu, N. R. Klymko, E. D. Perfecto, A. G. Merryman, K. A. Kelly, S. Purushothaman, S. J. Koester, R. Wisnieff, , and W. Haensch, "A 300-mm Wafer-Level Three-Dimensional Integration Scheme Using Tungsten Through-Silicon Via and Hybrid Cu-Adhesive Bonding," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2008.
- [37] J. V. Olmen, A. Mercha, G. Katti, C. Huyghebaert, J. V. Aelst, E. Seppala, Z. Chao, S. Armini, J. Vaes, R. C. Teixeira, M. V. Cauwenberghe, P. Verdonck., K. Verhemeldonck, A. Jourdain, W. Ruythooren, M. de Potter de ten Broeck, A. Opdebeeck, T. Chiarella, B. Parvais, I. Debusschere, T. Hoffmann, B. D. Wachter, W. Dehaene, M. Stucchi, M. Rakowski, P. Soussan, R. Cartuyvels, E. Beyne, S. Biesemans, and B. Swinnen, "3D Stacked IC Demonstration using a Through Silicon Via First Approach," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2008.
- [38] D. Chen, W. Chiou, M. Chen, T. Wang, K. Ching, H. Tu, W. Wu, C. Yu, K. Yang, H. Chang, M. Tseng, C. Hsiao, Y. Lu, H. Hu, Y. Lin, C. Hsu, W. S. Shue, and C. Yu, "Enabling 3D-IC foundry technologies for 28 nm node and beyond: through-silicon-via integration with high throughput die-to-wafer stacking," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2009.
- [39] L. D. Cioccio, P. Gueguen, T. Signamarcheix, M. Rivoire, D. Scevola, R. Cahours, P. Leduc, M. Assous, and L. Clavelier, "Enabling 3D Interconnects with Metal Direct Bonding," in *Proceedings IEEE International Interconnect Technology Conference*, pp. 152–154, June 2009.
- [40] G. Katti, A. Mercha, J. V. Olmen, C. Huyghebaert, A. Jourdain, M. Stucchi, M. Rakowski, I. Debusschere, P. Soussan, W. Dehaene, K. D. Meyer, Y. Travaly, E. Beyne, S. Biesemans, and B. Swinnen, "3D stacked ICs using Cu TSVs and Die to Wafer Hybrid Collective bonding," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2009.
- [41] J. Cong and Y. Zhang, "Thermal Via Planning for 3-D ICs," in *Proceedings IEEE/ACM International Conference on Computer-Aided Design*, pp. 745–752, Nov. 2005.
- [42] V. F. Pavlidis and E. G. Friedman, "Timing-driven via placement heuristics for three-dimensional ICs," in *Integration, the VLSI Journal*, vol. 41, pp. 489–508, July 2008.
- [43] D. Khalil, Y. Ismail, M. Khellah, T. Karnik, and V. De, "Analytical Model for the Propagation Delay of Through Silicon Vias," in *Proceedings International Symposium on Quality Electronic Design*, pp. 553–556, Mar. 2008.

- [44] I. Savidis and E. G. Friedman, "Closed-Form Expressions of 3-D Via Resistance, Inductance, and Capacitance," in *IEEE Transactions on Electron Devices*, vol. 56, pp. 1873–1881, Sept. 2009.
- [45] T. Bandyopadhyay, R. Chatterjee, D. Chung, M. Swaminathan, and R. Tummala, "Electrical Modeling of Through Silicon and Package Vias," in *Proceedings IEEE Conference on 3D System Integration*, Sept. 2009.
- [46] C. Xu, H. Li, R. Suaya, and K. Banerjee, "Compact AC Modeling and Analysis of Cu, W, and CNT based Through-Silicon Vias (TSVs) in 3-D ICs," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2009.
- [47] D. H. Kim and S. K. Lim, "Through-Silicon-Via-aware Delay and Power Prediction Model for Buffered Interconnects in 3D ICs," in *Proc. ACM/IEEE Int. Workshop on System Level Interconnect Prediction*, pp. 25–31, June 2010.
- [48] K. Yoon, G. Kim, W. Lee, T. Song, J. Lee, H. Lee, K. Park, and J. Kim, "Modeling and Analysis of Coupling between TSVs, Metal, and RDL interconnects in TSV-based 3D IC with Silicon Interposer," in *Proceedings IEEE Electronics Packaging Technology Conference*, pp. 702–706, Dec. 2009.
- [49] T. Sakurai and K. Tamaru, "Simple Formulas for Two- and Three-Dimensional Capacitances," in *IEEE Transactions on Electron Devices*, vol. 30, pp. 183–185, Feb. 1983.
- [50] A. Bansal, B. C. Paul, and K. Roy, "An Analytical Fringe Capacitance Model for Interconnects Using Conformal Mapping," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, pp. 2765–2774, Dec. 2006.
- [51] H. C. Ohanian and J. T. Markert, *Physics for Engineers and Scientists*. W. W. Norton & Company, 2007.
- [52] A. W. Topol, J. D. C. La Tulipe, L. Shi, D. J. Frank, K. Bernstein, S. E. Steen, A. Kumar, G. U. Singco, A. M. Young, K. W. Guarini, and M. Jeong, "Three-dimensional Integrated Circuits," in *IBM J. Res. & Dev.*, p. 491.
- [53] B. Goplen and S. Sapatnekar, "Placement of 3D ICs with Thermal and Interlayer Via Considerations," in *Proceedings IEEE/ACM Design Automation Conference*, pp. 626–631, June 2007.
- [54] P. Spindler, U. Schlichtmann, and F. M. Johannes, "Kraftwerk2 - A Fast Force-Directed Quadratic Placement Approach Using an Accurate Net Model," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1398–1411, Aug. 2008.
- [55] H. Yan, Z. Li, Q. Zhou, and X. Hong, "Via Assignment Algorithm for Hierarchical 3-D Placement," in *Proc. IEEE Int. Conf. on Communications, Circuits and Systems*, pp. 1225–1229, May 2005.

- [56] H.-H. S. Lee and K. Chakrabarty, “Test Challenges for 3D Integrated Circuits,” in *IEEE Design & Test of Computers*, pp. 26–35, Sept. 2009.
- [57] T. Thorolfsson, G. Luo, J. Cong, and P. D. Franzon, “Logic-on-Logic 3D Integration and Placement,” in *Proceedings IEEE Conference on 3D System Integration*, 2010.
- [58] M. Pathak, Y.-J. Lee, T. Moon, and S. K. Lim, “Through-Silicon-Via Management during 3D Physical Design: When to Add and How Many?,” in *Proceedings IEEE/ACM International Conference on Computer-Aided Design*, pp. 387–394, Nov. 2010.
- [59] X. He, S. Dong, Y. Ma, and X. Hong, “Simultaneous Buffer and Interlayer Via Planning for 3D Floorplanning,” in *Proceedings International Symposium on Quality Electronic Design*, 2009.
- [60] J. Knechtel, I. L. Markov, and J. Lienig, “Assembling 2D Blocks into 3D Chips,” in *Proceedings International Symposium on Physical Design*, pp. 81–88, 2011.
- [61] C. Chu and Y.-C. Wong, “FLUTE: Fast Lookup Table Based Rectilinear Steiner Minimal Tree Algorithm for VLSI Design,” in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, pp. 70–83, Jan. 2008.
- [62] X. Tang, R. Tian, and M. D. F. Wong, “Optimal Redistribution of White Space for Wire Length Minimization,” in *Proceedings Asia and South Pacific Design Automation Conference*, pp. 412–417, Jan. 2005.
- [63] E. Wong and S. K. Lim, “Whitespace Redistribution For Thermal Via Insertion In 3D Stacked ICs,” in *Proceedings IEEE International Conference on Computer Design*, pp. 267–272, Oct. 2007.
- [64] X. Li, Y. Ma, X. Hong, S. Dong, and J. Cong, “LP Based White Space Redistribution for Thermal Via Planning and Performance Optimization in 3D ICs,” in *Proceedings Asia and South Pacific Design Automation Conference*, pp. 209–212, Jan. 2008.
- [65] D. H. Kim, S. Kim, , and S. K. Lim, “Impact of Sub-micron Through-Silicon Vias on the Quality of Today and Future 3D IC Designs,” in *Proc. ACM/IEEE Int. Workshop on System Level Interconnect Prediction*, June 2011.
- [66] R. E. Farhane, M. Assous, P. Leduc, A. Thuai, D. Bouchu, H. Feldis, and N. Sillon, “A successful implementation of Dual Damascene architecture to copper TSV for 3D high density,” in *Proceedings IEEE Conference on 3D System Integration*, 2010.
- [67] M. Motoyoshi, “Through-Silicon Via (TSV),” in *Proceedings of the IEEE*, vol. 97, Jan. 2009.
- [68] M. Koyanagi, T. Fukushima, and T. Tanaka, “High-Density Through Silicon Vias for 3-D LSIs,” in *Proceedings of the IEEE*, vol. 97, Jan. 2009.
- [69] PTM, “Predictive Technology Model.” <http://ptm.asu.edu>.

- [70] ITRS, “International Technology Roadmap for Semiconductors 2007 Edition Interconnect.” <http://www.itrs.net>.
- [71] Y.-J. Lee and S. K. Lim, “Timing Analysis and Optimization for 3D Stacked Multi-Core Microprocessors,” in *Proc. Int. 3D System Integration Conference*, Nov. 2010.
- [72] D. H. Kim and S. K. Lim, “Impact of Through-Silicon-Via Scaling on the Wirelength Distribution of Current and Future 3D ICs,” in *Proceedings IEEE International Interconnect Technology Conference*, May 2011.
- [73] A. B. Kahng and K. Samadi, “CMP Fill Synthesis: A Survey of Recent Studies,” in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, pp. 3–19, Jan. 2008.
- [74] A. B. Kahng, K. Samadi, and R. O. Topaloglu, “Recent topics in CMP-related IC Design for Manufacturing,” in *Advanced Metallization Conference*, pp. 674–680, Oct. 2008.
- [75] H. Cai, *Modeling of Pattern Dependencies in the Fabrication of Multilevel Copper Metallization*. PhD thesis, Massachusetts Institute of Technology, June 2007.
- [76] A. B. Kahng, G. Robins, A. Sings, H. Wang, and A. Zelikovsky, “Filling and Slotting : Analysis and Algorithms,” in *Proceedings International Symposium on Physical Design*, pp. 95–102, Apr. 1998.
- [77] M. Cho, H. Xiang, R. Puri, and D. Z. Pan, “Wire Density Driven Global Routing for CMP Variation and Timing,” in *Proceedings IEEE/ACM International Conference on Computer-Aided Design*, pp. 487–492, Nov. 2006.
- [78] T.-C. Chen, M. Cho, D. Z. Pan, and Y.-W. Chang, “Metal-Density-Driven Placement for CMP Variation and Routability,” in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 2145–2155, Dec. 2008.
- [79] “OpenCores.” <http://www.opencores.org>.
- [80] “FreePDK45.” <http://www.eda.ncsu.edu/wiki/FreePDK>.

R.1 Related Publications

This dissertation is based on and/or related to the work and results presented in the following publications in print:

- [1] **D. H. Kim** and S. K. Lim, “Bus-Aware Microarchitectural Floorplanning,” in *Proceedings IEEE/ACM Asia South Pacific Design Automation Conference*, pp. 204–208, January 2008.
- [2] **D. H. Kim** and S. K. Lim, “Global Bus Route Optimization with Application to Microarchitecture,” in *Proceedings IEEE International Conference on Computer Design*, pp. 658–663, October 2008.
- [3] **D. H. Kim**, S. Mukhopadhyay, and S. K. Lim, “TSV-aware Interconnect Length and Power Prediction for 3D Stacked ICs,” in *Proceedings IEEE International Interconnect Technology Conference*, pp. 26–28, June 2009.
- [4] **D. H. Kim**, S. Mukhopadhyay, and S. K. Lim, “Through-Silicon-Via Aware Interconnect Prediction and Optimization for 3D Stacked ICs,” in *Proceedings ACM/IEEE International Workshop on System Level Interconnect Prediction*, pp. 85–92, July 2009.
- [5] **D. H. Kim**, K. Athikulwongse, and S. K. Lim, “A Study of Through-Silicon-Via Impact on the 3D Stacked IC Layout,” in *Proceedings International Conference on Computer-Aided Design*, pp. 674–680, November 2008.
- [6] **D. H. Kim** and S. K. Lim, “Through-Silicon-Via-aware Delay and Power Prediction Model for Buffered Interconnects in 3D ICs,” in *Proceedings ACM/IEEE International Workshop on System Level Interconnect Prediction*, pp. 25–31, June 2010.
- [7] **D. H. Kim**, Y.-K. Wu, R. O. Topaloglu, and S. K. Lim, “Enabling 3D Integration Through Optimal Topography,” in *Proceedings IEEE International Workshop on Design for Manufacturability & Yield*, pp. 70–73, June 2010.
- [8] M. Cho, C. Liu, **D. H. Kim**, S. K. Lim, and S. Mukhopadhyay, “Design Method and Test Structure to Characterize and Repair TSV Defect Induced Signal Degradation in 3D System,” in *Proceedings IEEE/ACM International Conference on Computer-Aided Design*, pp. 694–697, November 2010.
- [9] M. B. Healy, K. Athikulwongse, R. Goel, M. M. Hossain, **D. H. Kim**, Y.-J. Lee, D. L. Lewis, T.-W. Lin, C. Liu, M. Jung, B. Ouellette, M. Pathak, H. Sane, G. Shen, D. H. Woo, X. Zhao, G. H. Loh, H.-H. S. Lee, and S. K. Lim, “Design and Analysis of 3D-MAPS: A Many-Core 3D Processor with Stacked Memory,” in *Proceedings IEEE Custom Integrated Circuits Conference*, September 2010.
- [10] M. Bashir, **D. H. Kim**, S. K. Lim, and L. Milor, “TDDB Chip Reliability in Copper Interconnects,” in *Proceedings IEEE International Integrated Reliability Workshop*, pp. 121–124, October 2010.

- [11] M. Bashir, L. Milor, **D. H. Kim**, and S. K. Lim, "Methodology to Determine the Impact of Linewidth Variation on Chip Scale Copper/Low-k Backend Dielectric Breakdown," in *Microelectronics Reliability*, Vol. 50, Issue 9–11, pp. 1341–1346, 2010.
- [12] **D. H. Kim**, S. Mukhopadhyay, and S. K. Lim, "Fast and Accurate Analytical Modeling of Through-Silicon-Via Capacitive Coupling," in *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, Vol. 1, No. 2, pp. 168–180, 2011.
- [13] T. Song, C. Liu, **D. H. Kim**, J. Cho, J. Kim, J. S. Pak, S. Ahn, J. Kim, K. Yoon, and S. K. Lim, "Analysis of TSV-to-TSV Coupling with High-Impedance Termination in 3D ICs," in *Proceedings IEEE International Symposium on Quality Electronic Design*, pp. 122–128, March 2011.
- [14] M. Bashir, **D. H. Kim**, K. Athikulwongse, S. K. Lim, and L. Milor, "Backend Low-k TDDDB Chip Reliability Simulator," in *Proceedings IEEE International Reliability Physics Symposium*, pp. 2C.2.1–2C.2.10, April 2011.
- [15] **D. H. Kim** and S. K. Lim, "Impact of Through-Silicon-Via Scaling on the Wirelength Distribution of Current and Future 3D ICs," in *Proceedings IEEE International Interconnect Technology Conference*, May 2011.
- [16] **D. H. Kim**, S. Kim, and S. K. Lim, "Impact of Sub-micron Through-Silicon Vias on the Quality of Today and Future 3D IC Designs," in *Proceedings ACM/IEEE International Workshop on System Level Interconnect Prediction*, June 2011.
- [17] **D. H. Kim** and S. K. Lim, "A Study on the Impact of Nano-Scale TSVs on 3D IC Designs," in *SRC TECHCON Conference*, September 2011.
- [18] M. Bashir, L. Milor, **D. H. Kim**, and S. K. Lim, "Impact of Irregular Geometries on Low-k Dielectric Breakdown," in *Microelectronics Reliability*, Vol. 51, Issue 9–11, pp. 1582–1586, 2011.
- [19] M. Cho, C. Liu, **D. H. Kim**, S. K. Lim, and S. Mukhopadhyay, "Pre-bond and Post-bond Test and Signal Recovery Structure to Characterize and Repair TSV Defect Induced Signal Degradation in 3D System," in *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, Vol. 1, No. 11, pp. 1718–1727, 2011.
- [20] **D. H. Kim**, R. O. Topaloglu, and S. K. Lim, "TSV Density-driven Global Placement for 3D Stacked ICs," in *Proceedings International SoC Design Conference*, November 2011.
- [21] **D. H. Kim**, R. O. Topaloglu, and S. K. Lim, "Block-level 3D IC Design with Through-Silicon-Via Planning," in *Proceedings IEEE/ACM Asia South Pacific Design Automation Conference*, January 2012.
- [22] **D. H. Kim**, K. Athikulwongse, M. B. Healy, M. M. Hossain, M. Jung, I. Khorosh, G. Kumar, Y.-J. Lee, D. L. Lewis, T.-W. Lin, C. Liu, S. Panth, M. Pathak, M. Ren, G.

- Shen, T. Song, D. H. Woo, X. Zhao, J. Kim, H. Choi, G. H. Loh, H.-H. S. Lee, and S. K. Lim, “3D-MAPS: 3D Massively Parallel Processor with Stacked Memory,” in *Proceedings IEEE International Solid-State Circuits Conference*, February 2012.
- [23] K. Yang, **D. H. Kim**, and S. K. Lim, “Design Quality Tradeoff Studies for 3D ICs Built with Nano-scale TSVs and Devices,” in *IEEE International Symposium on Quality Electronic Design*, March 2012.
- [24] C.-C. Chen, M. Bashir, L. Milor, **D. H. Kim**, and S. K. Lim, “Backend Dielectric Chip Reliability Simulator for Complex Interconnect Geometries,” in *Proceedings IEEE International Reliability Physics Symposium*, April 2012.